

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problem Mailbox.**



*Sign*

PATENTS

IN THE UNITED STATES PATENT & TRADEMARK OFFICE

In Re Application of:

Manfred Jonsson

Serial No.: 10/689,473

Filed: October 20, 2003

For: PORTABLE SET-UP DEVICE AND )  
METHOD FOR A COIN HANDLING )  
OR VALUABLE PAPER HANDLING )  
MACHINE )

Examiner: NOT YET ASSIGNED

Docket No.: S247 1040.1

**SUBMISSION OF PRIORITY DOCUMENT  
CLAIM OF PRIORITY**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir:

Applicant submits herewith a certified copy of the priority document in the above case (Swedish Patent Application No. 0203131-8, filed October 21, 2002).

Applicant hereby restates its claim of priority and does hereby claim the benefit of the filing date of prior Swedish Patent Application No. 0203131-8, filed October 21, 2002, pursuant to the provisions of 35 U.S.C. § 119.

Applicant courteously requests that the claim of priority and the submitted priority document be accepted to perfect Applicant's claim of priority.

Respectfully submitted,

Louis T. Isaf  
Reg. No. 29,078

Date: May 14, 2004

Womble Carlyle Sandridge & Rice  
P.O. Box 7037  
Atlanta, Georgia 30357-0037  
Tel: (404) 962-7523  
Fax: (404) 870-8173



CERTIFICATE OF MAILING

I hereby certify that this correspondence is being mailed in an envelope by First Class mail to:

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

on 5-14-04

Suzanne Skinner  
(Typed or printed name of person mailing correspondence)

Suzanne Skinner  
(Signature of person mailing correspondence)

Serial Number: 10/689,473

Filing Date: October 20, 2003

Title: PORTABLE SET-UP DEVICE AND METHOD FOR A COIN  
HANDLING OR VALUABLE PAPER HANDLING MACHINE

Our Reference Number: S247 1040.1

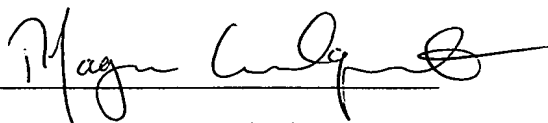
Submission of Priority Document – Claim of Priority  
Certified Priority Document  
Return Postcard



## DECLARATION

I, the undersigned Magnus Lundquist, technical translator, of Bellevuevägen 46, Malmö, Sweden, do hereby declare that I am well versed in the English and Swedish languages and that the document annexed hereto is a true and complete translation of the Official Document also annexed concerning the Swedish Patent Application No. 0203131-8 filed on the 21st of October 2002 by Scan Coin Industries AB, Malmö, Sweden.

Signed this 30th day of October 2003

  
Magnus Lundquist



THE SWEDISH PATENT AND REGISTRATION OFFICE  
Patent Department

## Certificate

*This is to certify that the annexed documents are true copies of the documents originally filed with the Swedish Patent and Registration Office in the following Application.*

(Seal of  
the Patent  
Office)

*Applicant(s)*

*Scan Coin Industries AB, Malmö SE*

*Patent Application No. 0203131-8*

*Filing Date*

*21 October 2002*

*Stockholm, 22 September 2003*

*For the Patent and Registration Office*

*Hjördis Segerlund  
Hjördis Segerlund*

*Fee SEK 170*

PORTABLE SET-UP DEVICE AND METHOD FOR A COIN HANDLING OR  
VALUABLE PAPER HANDLING MACHINE

Field of the Invention

The present invention relates to machines for handling coins or valuable papers and, more specifically, to  
5 the configuration of such machines.

Background Art

It goes without saying that machines of the kind stated above are known in the art. Examples thereof are  
10 coin sorters, coin counters, banknote counters, banknote readers, check readers, vending machines, gambling machines, ATMs, machines for testing the quality of coins, banknotes or checks, machines for identifying counterfeit or foreign coins/banknotes/checks, etc. Such  
15 machines usually comprise a large number of operating parameters, settings and the like, which can be referred to collectively as configuration data. Although a particular machine is configured at the factory, subsequent reconfiguring thereof at the site of operation may be  
20 necessary, for different reasons, on one or more occasions. Reconfiguring may include the need to update the software of the machine.

One example of a prior-art machine of the kind stated above is shown in Fig. 1 in the form of a coin sorter  
25 100. The coin sorter 100 comprises a coin intake associated with a hopper-type coin feeding device 106 as well as a coin discriminator 108, a sorting device 110, a coin return means 112 and a plurality of coin cassettes 114. The machine 100 further comprises an internal microprocessor (CPU) or similar control unit, which in turn is  
30 connected to a memory, such as a RAM, ROM, EEPROM, flash memory, hard disk, or any combination thereof. The memory

stores, inter alia, machine software to be executed by the microprocessor, and can also store configuration and working data for the coin discriminator 108 and the sorting device 110. One of the tasks of the software is to  
5 form a user interface together with a front display 102 and keypad 104.

Fig. 1 illustrates a prior-art method of configuring the coin sorter 100 to change, for example, its settings and/or update its software. A service engineer brings a  
10 portable computer 120 to the site of operation where the machine 100 is used. The computer 120 is connected to the machine 100 by means of, for example, a serial cable 124 between an RS232 interface on the computer and a corresponding RS232 interface on the machine. It may also be  
15 necessary for the service engineer to connect the computer 120 to the power grid through a power supply unit 122 to ensure that the service can be carried out even if the computer battery runs out. When the computer is installed, the service engineer first launches its operating system, for example Microsoft® Windows, and then  
20 an appropriate configuration application. Using the application the service engineer is then able to configure the machine 100. To use the configuration application in this way the service engineer must, of course,  
25 be qualified in the use thereof.

Examples of configuration data for the coin sorter 100 are help texts and menus for the display 102, or the equivalent for a printer installed on the machine (not shown in Fig. 1). Other examples are parameter values for  
30 the coin discriminator 108 - such as different acceptance ranges within which a number of physical parameters relating to different coin denominations must fall for the coins to be considered valid. Such physical parameters can include, for example, conductivity, permeability,  
35 thickness, diameter or weight. One of many situations in which the machine 100 may need to be reconfigured is when it is to be used for a different currency,

or when the coin system which the machine is set to handle is changed in any way.

The machine's software may need to be updated due to errors, performance deficiencies, or the like, or when  
5 the user has requested a new or additional functionality.

One problem with, and drawback of, the current configuration method is that it requires a qualified service engineer, who has to travel to the site of operation bringing a relatively heavy and bulky PC (including  
10 cables and power supply unit). Furthermore, connecting the computer and launching its operating system and the relevant application is time-consuming. Moreover, a sufficiently large working area is needed to set up the computer 120 adjacent to the machine 100.

15

#### Summary of the Invention

One object of the present invention is to solve or at least alleviate the above problems and drawbacks. This object has been achieved, in brief, by the creation of a  
20 new type of configuration tool, viz. a portable device which is intended to be connected, in a first step, to a computer comprising said configuration application. In this position, the device emulates the machine for handling coins or valuable papers, and all settings made by  
25 the user of the configuration application are buffered in the portable device. At a later stage, the portable device is connected to the machine and emulates, in this position, the configuration application, the settings buffered in the portable device being executed on the  
30 machine. Similarly, the portable device can be used to update existing software in the machine.

By using this new configuration tool, i.e. a portable device according to the invention, a number of advantages are obtained. First, the configuration method  
35 is considerably simplified and, thus, the level of training required at the site of operation is reduced. It is no longer necessary for the highly qualified service



engineer to travel to the site of operation, instead he or she can configure the portable device at his or her usual place of work. The device can then be delivered to the site of operation, where a non-qualified user can  
5 connect it to the appropriate machine, following which the configuration is carried out automatically. One advantage in this conjunction is that the portable device is suitably miniaturized into a very convenient format. Furthermore, the service time required at the site of  
10 operation is reduced, since it is no longer necessary to connect and launch a configuration computer.

More specifically, the above stated object is achieved by a portable device and a method according to the independent claims.

15 Thus, a first aspect of the invention is a portable device for configuring a machine of the kind which handles coins or valuable papers and which has an interface for connecting an external unit, thereby allowing the machine to be configured from the external unit. The  
20 portable device has a control unit; a memory; a first interface adapted to be connected to said external unit; and a second interface adapted to be connected to the interface of said machine, the control unit being adapted, when the portable device is connected to the external  
25 unit via the first interface, to emulate said machine and to receive configuration data intended for said machine from the external unit and store it in the memory, and the control unit being adapted, when the portable device is connected to said machine via the second interface, to  
30 emulate the external unit and deliver said configuration data stored in the memory to said machine.

The term configuring here comprises setting different parameters, reference data, language support and the like in the machine. It also comprises updating the  
35 machine software. However, it is not crucial to the invention that both of these steps should be included in the term configuring. Correspondingly, the term confi-

guration data may comprise settings and/or parameters relating to validity/authenticity/denomination/type of coins and valuable papers, respectively; reference data; or language support data. Moreover, the term configura-  
5 tion data may comprise program code in machine-readable form, with which the software of the machine is to be updated.

The external unit may be a computer of the type including a portable computer, a personal computer, a  
10 handheld computer, a workstation or the like.

The first and second interfaces may be serial, for example according to the RS232 standard.

According to one embodiment, the control unit is further adapted, when the portable device is connected  
15 to said machine via the second interface, to receive operating data from said machine and store it in the memory and, when the portable device is connected to the external unit via the first interface, to transfer said operating data stored in the memory to the external unit.

20 Said operating data may comprise number/validity/authenticity/denomination/type of coins and valuable papers, respectively that have been handled by the machine, and/or physical parameters for such coins and valuable papers detected by the machine.

25 Power may be supplied to the portable device from the first and second serial interfaces. According to one embodiment, the first and the second interface form part of the same physical unit.

Advantageously, the machine described above can be a  
30 coin sorter or a coin counter.

A second aspect of the invention is a method of configuring a machine of the type which handles coins or valuable papers and which can be configured from an external unit. The method comprises the steps of connect-  
35 ing a portable device, other than said machine and said external unit, to said external unit; receiving configuration data intended for said machine from the external

unit and storing it in the portable device; connecting the portable device to said machine; and delivering said stored configuration data to said machine.

Further objects, advantages and features of the invention will appear from the following detailed description of the invention, the appended claims and the drawings.

#### Brief Description of the Drawings

10       The invention will now be described in more detail, reference being made to the accompanying drawings.

Fig. 1 illustrates a prior-art method of interconnecting a computer and a coin handling machine by means of a serial cable for the purpose of configuring the machine, for example to change its settings or update its software.

Fig. 2 is a perspective view of a portable device according to the invention for configuring the machine shown in Fig. 1, optionally also for recording operating data from said machine.

Fig. 3 illustrates the portable device of Fig. 2 connected to the computer shown in Fig. 1.

Fig. 4 is a rear view of the coin handling machine shown in Fig. 1.

25       Fig. 5 illustrates the portable device of Fig. 2 connected to the machine shown in Fig. 1.

Fig. 6 is a block diagram of main components of the coin handling machine shown in Figs 1, 4 and 5.

Fig. 7 is a block diagram of main components of the portable device shown in Figs 2, 3 and 5.

Fig. 8 is a block diagram of an arrangement for supplying power to the components shown in Fig. 7.

Fig. 9 is a flowchart of a main routine for the portable device according to Figs 2, 3, 5 and 7.

35       Fig. 10 is a flowchart of a routine for the portable device in the case where it is connected to the machine in the way illustrated in Fig. 5.

Fig. 11 is a flowchart of a routine for the portable device in the case where it is connected to the computer in the way illustrated in Fig. 3.

## 5 Detailed Description of the Invention

Figs 2-5 illustrate a typical use of a portable device 200 according to a preferred embodiment of the invention. As shown, the device 200 has the form of a small box based on an apparatus housing and having a  
10 first connector 210 at one end and a second connector 220 at the opposite end. In the preferred embodiment, the connectors 210, 220 are conventional 9-pin serial ports according to the RS232 standard. The first port 210 is adapted to be connected to a computer 120 (see Fig. 3),  
15 while the second port 220 is adapted to be connected to a corresponding interface 402 on a coin handling machine 100 (see Figs 4 and 5). In normal circumstances, the portable device 200 is not connected to both the computer 120 and the machine 100 at the same time, but first to  
20 the computer (in which state the device emulates the machine 100 and receives configuration data from the computer 120) and then to the machine (in which state the device emulates the computer and forwards said configuration data to the machine 100). However, the reverse  
25 procedure, i.e. first connecting it to the machine 100 and then to the computer 120 can be applied when using the device 200 to record operating data from the machine 100 for post-processing on the computer 120.

Fig. 6 is a block diagram of the main parts of a  
30 coin handling machine 600, more specifically the coin sorter 100 shown in Fig. 1. However, the machine 600 could just as well be a coin or banknote counter, a vending machine, a gambling machine, an ATM (Automatic Teller Machine), a machine for testing the quality of  
35 coins, banknotes or checks, a machine for identifying counterfeit or foreign coins/banknotes/checks, etc.

A quantity of coins to be sorted by the machine 600 are deposited in a coin intake 610. The coins are transported via a coin feeder 620, for example a hopper-type coin feeding device (cf. 106 in Fig. 1) or a conveyor belt, to a coin discriminator 630 (cf. 108 in Fig. 1). The coin discriminator 630 is operatively connected to a control unit 632 in the form of a microprocessor (CPU) or the like, which in turn is operatively connected to a memory 634, such as a RAM, ROM, EEPROM, flash memory, hard disk, or any combination thereof. At least parts of the memory 634 can be implemented by internal memory in the control unit. The control unit 632 is in charge of the overall mode of operation of the machine 600, including controlling a display 636 and a key pad 638, which form a user interface (cf. respectively 102 and 104 in Fig. 1). The configuration data relating to the machine which the portable device according to the invention is intended to update is stored in the memory 634. A configuration interface 639 (cf. the serial port 402 in Fig. 4) allows new/modified configuration data to be received by the control unit 632 from the portable device 200 and to be stored in the memory 634.

The coin handling machine 600 further comprises a coin return unit 640 (cf. 112 in Fig. 1), which returns non-accepted coins through an external opening in the machine 600, so that the user can get such coins back. It is the coin discriminator 630 that, in this context, determines if a coin is to be accepted or not by detecting different physical parameters, which are used to determine the type, denomination, currency, identity, authenticity or the like of the coin. Examples of physical parameters are conductivity, permeability, thickness, diameter or weight. After the coin discriminator 630 a sorting device 650 (cf. 110 in Fig. 1) is provided which uses the coin-sensing results from the coin discriminator 630 to sort the coin into a particular coin container (cf. 114 in Fig. 1) associated with a coin stor-

age. The coin containers of the coin storage 660 are advantageously accessible to the user from the outside.

The control unit 632 stores various operating data in the memory 634. One example of such data is coin statistics from the coin discriminator 630 - in the form of the number of counted coins of each denomination, the number of rejected coins, or, in greater detail, statistics relating to the detected physical coin parameters. The configuration interface 639 allows this operating data to be transferred - also during operation - to the portable device 200 for "recording", for example, coin statistics. The recorded coin statistics can then be downloaded, at a later stage, from the portable device 200 to the computer 120 for analysis or other post-processing operations.

The portable device 200 is shown in the form of a block diagram in Fig. 7. All the components of the block diagram will be described in more detail below following a brief summary thereof. A control unit 700 handles the overall operation of the device and controls the other components. A memory 710 is associated with the control unit 700 and serves as working memory and program memory therefor. The memory 710 is also used to buffer configuration data that is received from the computer 120 before it is transferred to the machine 100, as well as to buffer operating data recorded from the machine 100 before it is transferred to the computer 120. A set of light emitting diodes 720 (also designated 230, 232 in Fig. 2) are adapted to indicate different operational states and fault states to the user. A first communication unit 740 is connected to the first serial port 210, while a second communication unit 750 is connected to the second serial port 220.

Since the device 200 emulates the machine 100 in relation to the computer 120, and vice versa, the communication via the communication units 740 and 750, respectively, follows the same general serial communi-

cation standard that is used conventionally between the computer and the machine (i.e. the situation shown in Fig. 1). A brief specification of this serial communication standard is provided below.

- 5        To connect a machine to a computer, an RS232 cable is used with the following connections:

Computer			Machine	
25 pins	9 pins	Direction	9 pins	Signal
3	2	←	3	Tx
2	3	→	2	Rx
7	5	↔	5	GND
4	7	→	8	CTS (PC not ready)
5	8	←	4	RTS (Machine not ready)

When the machine is started it has the following  
10 default settings: 9600 baud, no parity, 8 data bits and 1 stop bit. Usually, the baud rate is then reset to 115200 baud.

The handshake signals CTS (Clear To Send) and RTS (Request To Send) are used to control the flow. RTS is  
15 used by the machine to indicate when it is not ready to receive data from RX. If anything is sent while the RTS is active, a communication error may occur. To avoid this, the computer is capable of telling the machine that it is not permitted to send anything. This is done  
20 by setting the CTS active.

A configuration or "setup" software protocol is used for communication between the computer and the machine. Such a protocol is generally designated PCcom. All messages sent between the computer and the machine has the  
25 same structure:

Direction	Data
PC - Machine	
→	Escape characters (ESC 27)
→	Command characters ('A' - 'Z')
→	Sub-command characters ('a' - 'z' or 'A' - 'Z')

↔ Data bytes  
↔ CSUM, one-byte checksum  
← ACK (6) if CSUM is OK  
NAK (21) if CSUM is not OK  
EM (25) if CSUM is OK and there is not  
enough data memory.

Referring again to Fig. 7 and to Fig. 8, the portable device 200 will now be described in more detail. Its mode of operation will then be described with reference to Figs 9-11.

Power is supplied to the portable device 200 from the machine 100 and the computer 120, respectively, via the outputs on the serial ports 210, 220, as shown in Fig. 8. In normal circumstances, serial ports are used for communication purposes only, and it is therefore important not to destroy the communication signals. To separate the communication signal 802 from the power supply, diodes 804 are provided, the characteristics of which are such as to allow current to flow in one direction only. By using a voltage multiplier 806 (e.g. National Semiconductor LMC7660) a positive voltage can be obtained from a negative voltage for power supply 808 to the components of the device. The power supply 808 can comprise, inter alia, a voltage regulator, such as ST Microelectronics LE33ABZ, for conversion to a voltage level suitable for the components, for example 3.3 V.

The voltage range of the COM port according to the RS232 specification is  $\pm 25$  V. However, this voltage range is not the same for all machines. Protective resistors may therefore be required on the input side.

The control unit 700 can be implemented by means of a microcontroller MSP430F149 from Texas Instruments. This microcontroller has a very low current consumption. It consumes only 250  $\mu$ A per MIPS at a supply voltage of 2.2 V and is designed primarily for applications with stringent requirements for a low power consumption. The supply-voltage range is 1.8-3.6 V. At a working frequency



of 5 MHz the total current consumption for the microcontroller is about 1300  $\mu$ A. The maximum working frequency is 8 MHz at 3.6 V. The architecture is based on a 16-bit RISC core with 27 instructions. Each instruction takes  
5 only one clock cycle. Thus, the instruction cycle time at 5 MHz is 0.2  $\mu$ s.

The microcontroller has a 32-60 KB flash-type program memory and a 1-2 KB RAM-type working memory. Furthermore, it has an AD-converter, which is advantageously  
10 used for voltage monitoring and temperature measurement.

One advantage of the microcontroller MSP430F149 and its siblings in the 14x series is that it has two USART circuits (Universal Synchronous Asynchronous Receive Transmit). Thus, in the preferred embodiment the communication units 740 and 750 are implemented by means of such  
15 USART circuits, i.e. they are physically incorporated in the control unit 700.

It goes without saying that the control unit 700 can be implemented by means of a component other than the one  
20 proposed above, for example Atmel ATMEGA161L 8PI 0143, Atmel T89C51RD2 or Microchip PIC18LF452.

For the implementation of the memory 710, flash memories programmed in parallel or EEPROMs programmed serially may suitably be considered. The advantage of  
25 EEPROMs is that their current consumption when writing is very low compared with flash memories. Flash memories use so-called sectors. Since a sector in the flash memory must be erased before writing can take place, the sector has to be fetched to the control unit RAM. This means  
30 that the RAM memory must be at least as large as the sector size of the flash memory. In the preferred embodiment, a flash memory of the type SST SST28VF040A was selected. The capacity is 512 KB and the supply-voltage range is 2.7-3.6 V. In the preferred embodiment, two physical memory modules are used, which gives a total capacity of 1024 KB. The memory access time is 150-200 ns,  
35 which is less than the access time of the selected con-

trol unit. The memory can be overwritten 100 000 times,  
and the information will remain for 100 years. Other  
conceivable memories 710 are, for example, Microchip  
24LC515, Microchip 25LC640, ST Microelectronics M95256-V  
5 or Atmel AT29BV040A.

At least parts of the memory 710 can be implemented  
by means of internal memory in the control unit, for  
example the program memory (flash) and the working memory  
(RAM) in the microcontroller MSP430F149.

10 The portable device is operated by software as will  
now be described. The mode of operation of the software  
is illustrated in Fig. 9 and is based on the principle  
of throughput. Following an initialisation, the program  
enters a "superloop". The program makes several thousands  
15 of cycles every second. The cycle velocity is dependent  
on what happens at a particular moment. The superloop  
must be faster than the limiting factor, which in this  
case is the speed of communication. The program could be  
regarded as a kind of state machine.

20 Each "block" always returns to the superloop and,  
thus, never locks the execution in infinite loops. One  
exception is if an error in the external memory is  
detected during initialization. In this case, it is not  
convenient for the initialization to continue since the  
25 remaining program code is dependent on the function of  
the external memory.

The task of the superloop is to monitor, for each  
cycle, the state of the different software blocks. The  
superloop is the mastermind of the software and is at the  
30 top of the hierarchy. For instance, it decides when it is  
time to write to the external memory or when it is time  
to communicate. To carry out its task it relies on soft-  
ware buffers, time measurement functions, interrupt func-  
tions, function pointers and communication units. There  
35 are also monitor clocks that continuously check that  
nothing is wrong.

Information from the computer is received by means of an interrupt. The interrupt is stored in a 20 byte intermediate buffer. A circular write-pointer owned by the interrupt is used to place data in the intermediate buffer. Circular here means that instead of always beginning storing from address zero, it continues storing from where it is currently positioned. When it reaches the end it moves back to position zero. For each new data package a counter is incremented by one. The counter keeps track of how much is stored in the intermediate buffer and is owned by both the interrupt and the super loop. For each cycle, the counter is checked by the superloop. When the counter differs from zero, a PCcom function is activated (cf. what has been previously described in this document). The PCcom interprets the information by means of a read-pointer and decides whether it is part of an ESC command or data. Data is placed by means of another write pointer in a data buffer with a capacity of 129 bytes. The read-pointer is cyclic and is owned by the superloop. However, the write-pointer for the data buffer is not cyclic, but starts from position zero with each new storing sequence. This pointer is owned by the superloop. The data buffer now contains data that is ready to be transferred by the PCcom. The PCcom can point directly at the desired data byte.

To simplify programming, two separate software buffers are used, one for each interrupt. The software buffers have been allocated the minimum size that is possible without impairing communication. The reason is that they take up space in the RAM memory, which is only, for example, 2 KB. Since the computer transmits a maximum of 129 bytes per sequence, this is an appropriate size of the PCcom data buffer.

The same concept is used for reception from the machine 100 as for reception from the computer 120. The difference is that the portable device 200 uses another intermediate buffer and another data buffer. These buf-

fers are of a different size. The intermediate buffer size is 15 bytes and the data buffer size is 10 bytes.

A common software buffer with a size of 920 bytes is used for transmission to the computer 120 and the machine  
5 100. The transmission is not controlled by interrupts, but is handled by the superloop. The superloop keeps track of where data is to be sent by means of flags.

The external memory management, i.e. managing data transfer to and from the external memory, is carried out  
10 by means of functions. To write to the memory, a write function is called with data and address as input parameters. For reading, a read function is called with address as input parameter, and data is returned.

The two memory modules are combined to form a block  
15 using low-level software. From the outside this is seen as a large continuous memory with linear addressing from 0x00000-0xFFFFF, i.e. 1 MB.

Example:

Bytes a and b are programmed to the addresses x and  
20 y (a and b are placed in different sectors). The following occurs:

1) The sector containing the address x is copied from the external memory to the memory management buffer in the microcontroller RAM. This does not take place if  
25 the sector in question is already present in the RAM.

2) Byte a is placed at address x in the memory management buffer.

3) The sector containing address x in the external memory is erased.

30 4) The memory management buffer is programmed to the current sector in the external memory.

5) The sector containing address y is copied from the external memory to the memory management buffer.

35 6) Byte b is placed at address y in the memory management buffer.

Byte b is not programmed to the external memory until a sector change or "timeout" occurs.

The program flow according to Fig. 9 starts with an initialization (steps 900-920) before the superloop is entered (steps 922-938). During initialization the following takes place:

The program begins by initialising all global variables in step 902. The external clock frequency (X1) of the CPU is set until the oscillator has been stabilized. The internal clock (DCO) is set to 500 kHz. The direction of the ports is set.

The program then checks whether the external flash memory is OK in step 904. If the memory is not OK an error message is indicated in step 906 by means of the light emitting diodes 720/230-232. However, if the memory is OK the program carries out the last initialisations in step 908:

The AD converter is set to convert 15 times per second. A timer is initialised so that it generates an interrupt every 100 ms. The transfer speed is set to 9600 baud. The real time clock (RTC) is set to an arbitrary time and date.

The global interrupt is then enabled in step 910. In step 912, the program checks whether there is any configuration data stored in the external memory. If not, a basic configuration is programmed to the memory in step 914.

In steps 916 and 918, the machine and the computer are given permission to send. Finally, the RTC timer is started in step 920. This completes the initialization part.

In the superloop, the RTC is updated in step 922. In steps 924 and 926, a check is then carried out to determine whether a timeout has occurred and which message is to be displayed. The program then checks, in step 928, for anything connected to the communication units. The following three cases are possible:

- Only the computer is connected (step 932, Fig. 11).
  - Only the machine is connected (step 930, Fig. 10).
  - Both the computer and the machine are connected
- 5 (steps 934-936).

When only the computer is connected, the following takes place, as shown in Fig. 11. The UART driver of the machine is turned off in step 1100, and the UART driver of the computer is turned on in step 1102. In step 1104,  
10 a check is carried out to determine whether any new character has been received from the computer. If so, the PCcom is run in step 1106. This step is designed to interpret the incoming data from the configuration program of the computer in the same way as the machine would  
15 have done if the machine had been connected directly to the computer. A check is then carried out, in step 1108, to determine if any character has to be transmitted to the computer. Finally, a check is carried out in step 1110 to determine whether the computer has requested a  
20 change of transfer speed.

When only the machine is connected, the following takes place, as shown in Fig. 10. To simplify the use of the portable device, different tasks are allocated and stored in "status\_mode". The program checks "status\_mode"  
25 and carries out the task in question depending on the current "status\_mode". There are, in all, eight different ways of combining "status\_mode":

- No "status\_mode" selected (step 1002, 1004): The execution is terminated. Otherwise, the communication  
30 unit 740 connected to the computer is turned off in step 1006, while the communication unit 750 connected to the machine is turned on in step 1008.

- Update the machine software (step 1010, 1012): Machine logon according to a predefined program updating protocol. After the update, the machine software is  
35 relaunched in step 1014. If "status\_mode" indicates anything in addition to program update (which is checked

in step 1015), the execution proceeds with a machine logon according to a predefined configuration protocol (step 1016).

- Upload configuration data: Update configuration data in the machine (steps 1018, 1020 and 1024, respectively).

- Data recorder: Record operating data from the machine and store it in external flash memory (step 1018, 1022 and 1026, respectively). Recorded operating data can then be transferred to the computer for analysis or the like.

When both the computer and the machine are connected, both communication units 740, 750 are turned on in steps 934 and 936. In this state, the portable device 200 is transparent to the computer and the machine. Its task is only to store important information and forward all the incoming information.

The invention has been described above by means of two exemplary embodiments. However, the invention is not in any way limited to these embodiments, but includes many variants within the scope of the invention as defined by the appended claims, as is readily realized by a person skilled in the art. Accordingly, one alternative is to use the same physical communication port and/or communication unit of the portable device for the connection to the computer as well as for the connection to the machine. In such an embodiment, the control unit of the portable device will be adapted to recognize which of the computer and the machine that is currently connected, and to act on the basis thereof.

## CLAIMS

1. A portable device (200) for configuring a machine (100) of the type which handles coins or valuable papers and which has an interface (402) for connecting an external unit (120), thereby allowing the machine to be configured from the external unit, the portable device being characterized by
- a control unit (700);
  - 10 a memory (710);
  - a first interface (210, 740) adapted to be connected to said external unit (120); and
  - a second interface (220, 750) adapted to be connected to the interface (402) of said machine (100);
  - 15 the control unit being adapted, when the portable device is connected to the external unit via the first interface, to emulate said machine and to receive configuration data intended for said machine from the external unit and store it in the memory, and
  - 20 the control unit being adapted, when the portable device is connected to said machine via the second interface, to emulate the external unit and deliver said configuration data stored in the memory to said machine.
2. A portable device as claimed in claim 1, wherein the external unit (120) is a computer of the type including a portable computer, a personal computer, a handheld computer, a workstation or the like.
3. A portable device as claimed in claim 1 or 2, wherein the first and second interfaces (210, 740, 220, 750) are serial.
- 30 4. A portable device as claimed in any one of the preceding claims, wherein said configuration data comprises at least one of the following: settings and/or parameters relating to
- 35 validity/authenticity/denomination/type of coins and



valuable papers, respectively; reference data; or language support data.

5 5. A portable device as claimed in any one of the preceding claims, wherein said configuration data comprises software adapted to be executed in a microprocessor (632) incorporated in the machine (100).

6. A portable device as claimed in any one of the preceding claims,

10 wherein the control unit (700) is adapted, when the portable device (200) is connected to said machine (100) via the second interface (210, 750), to receive operating data from said machine and store it in the memory (710); and

15 wherein the control unit (700) is adapted, when the portable device (200) is connected to the external unit (120) via the first interface (210, 740), to transfer said operating data stored in the memory to the external unit.

20 7. A portable device as claimed in claim 6, wherein said operating data comprises number/validity/authenticity/denomination/type of coins and valuable papers, respectively, that have been handled by the machine (100), and/or physical parameters detected by the machine for such coins and valuable papers.

25 8. A portable device as claimed in claim 3, wherein power is supplied to the device from the first and second serial interfaces (210, 740, 220, 750).

30 9. A portable device as claimed in any one of the preceding claims, wherein the first and the second interface form part of the same physical unit.

10. A portable device as claimed in any one of the preceding claims, wherein said machine (100) is a coin sorter or a coin counter.

35 11. A method of configuring a machine (100) of the type which handles coins or valuable papers and which can be configured from an external unit (120), the method being characterized by the steps of

connecting a portable device (200), other than said machine (100) and said external unit (120), to said external unit (120);

receiving configuration data intended for said  
5 machine from the external unit and storing it in the portable device;

connecting the portable device to said machine; and  
delivering said stored configuration data to said machine.

10 12. A method as claimed in claim 11, wherein the external unit (120) is a computer of the type including a portable computer, a personal computer, a handheld computer, a workstation or the like.

15 13. A method as claimed in claim 11 or 12, wherein said configuration data comprises at least one of the following: settings and/or parameters relating to validity/authenticity/denomination/type of coins and valuable papers, respectively; reference data; or language support data.

20 14. A method as claimed in any one of claims 11-13, wherein said configuration data comprises software adapted to be executed in said machine (100).

15. A method as claimed in any one of claims 11-14, further comprising the steps of  
25 connecting the portable device (200) to said machine (100);

receiving operating data from said machine and storing it in the portable device;

30 connecting the portable device to the external unit; and

transferring said stored operating data to the external unit.

16. A method as claimed in claim 15, wherein said operating data comprises number/validity/authenticity/  
35 denomination/type of coins and valuable papers, respectively, that have been handled by the machine (100) and/

or physical parameters detected by the machine for such coins and valuable papers.

17. A method as claimed in any one of claims 11-16,  
wherein said machine (100) is a coin sorter or a coin  
5 counter.

## ABSTRACT

A portable device (200) is provided for configuring a machine (100) of the type which handles coins or valuable papers and which has an interface (402) for connecting an external unit (120), thereby allowing the machine to be configured from the external unit. The portable device has a control unit (700); a memory (710); a first interface (210, 740) adapted to be connected to said external unit (120); and a second interface (220, 750) adapted to be connected to the interface (402) of said machine (100). The control unit is adapted, when the portable device is connected to the external unit via the first interface, to emulate said machine and to receive configuration data intended for said machine from the external unit and store it in the memory. The control unit is further adapted, when the portable device is connected to said machine via the second interface, to emulate the external unit and deliver said configuration data stored in the memory to said machine.

Elected for publication: Fig. 7

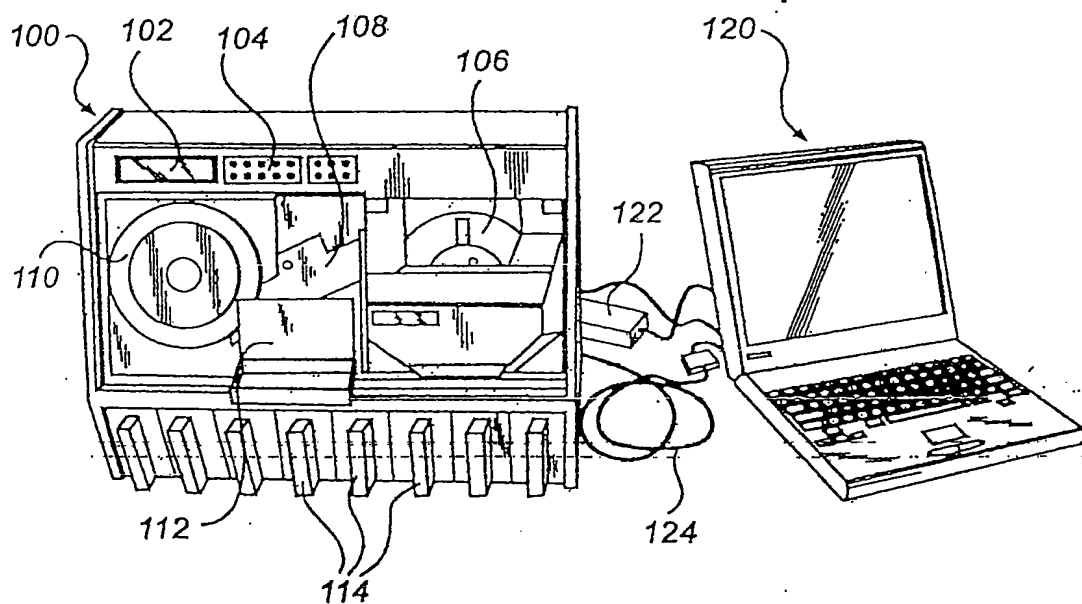
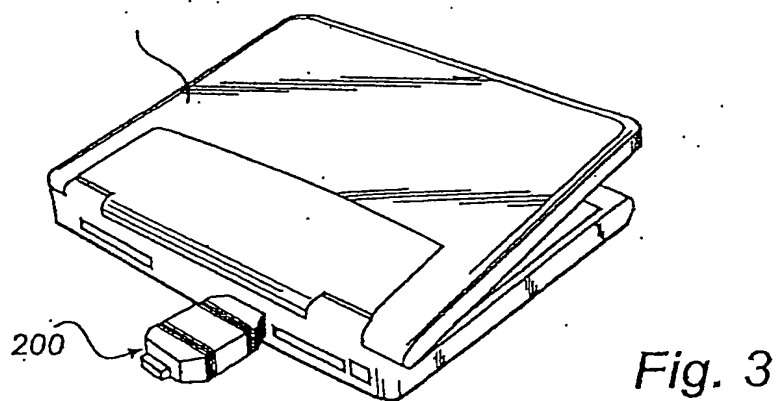
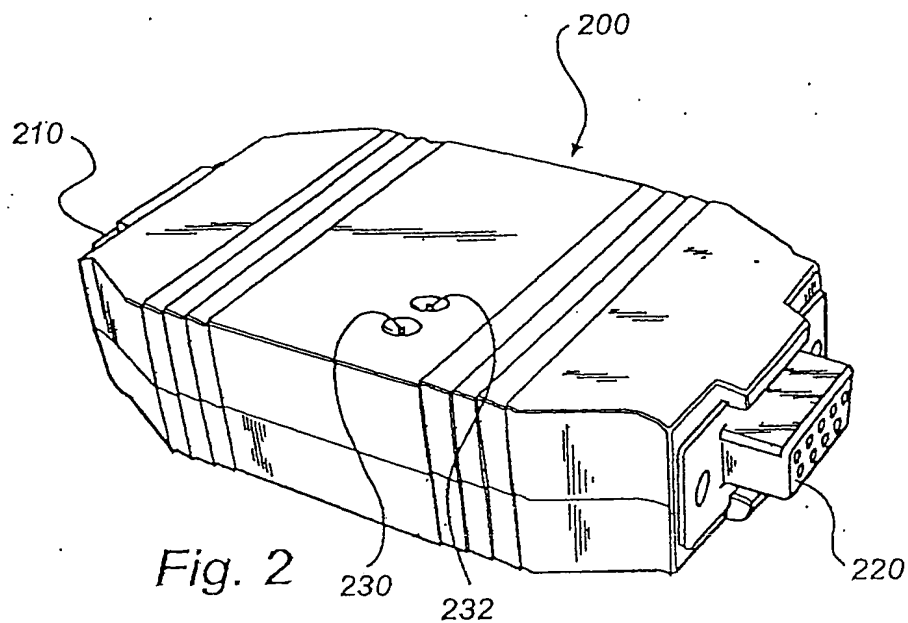
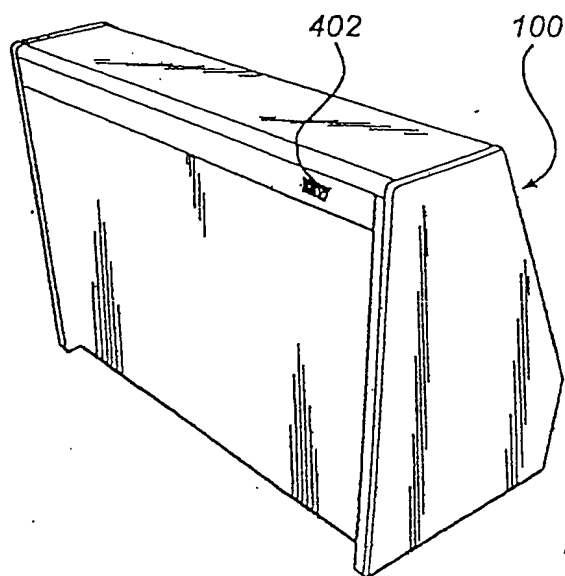


Fig. 1

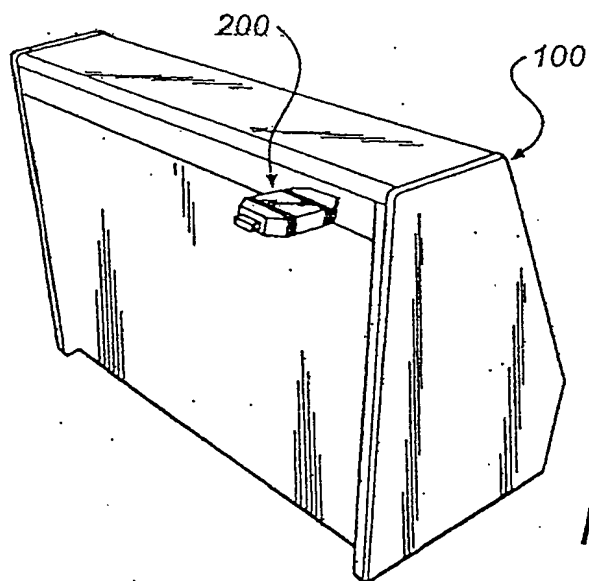
2/8



3/8

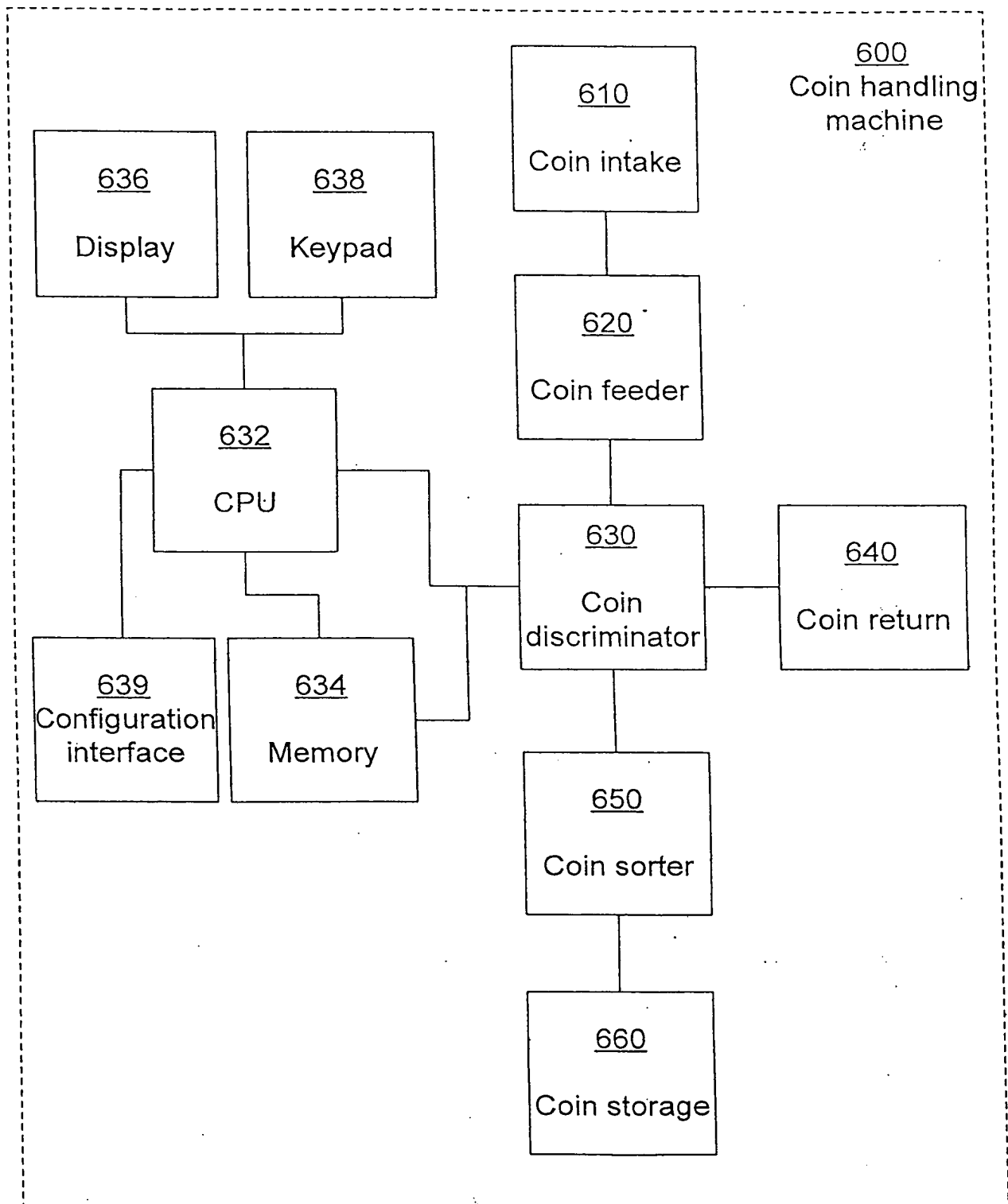


*Fig. 4*



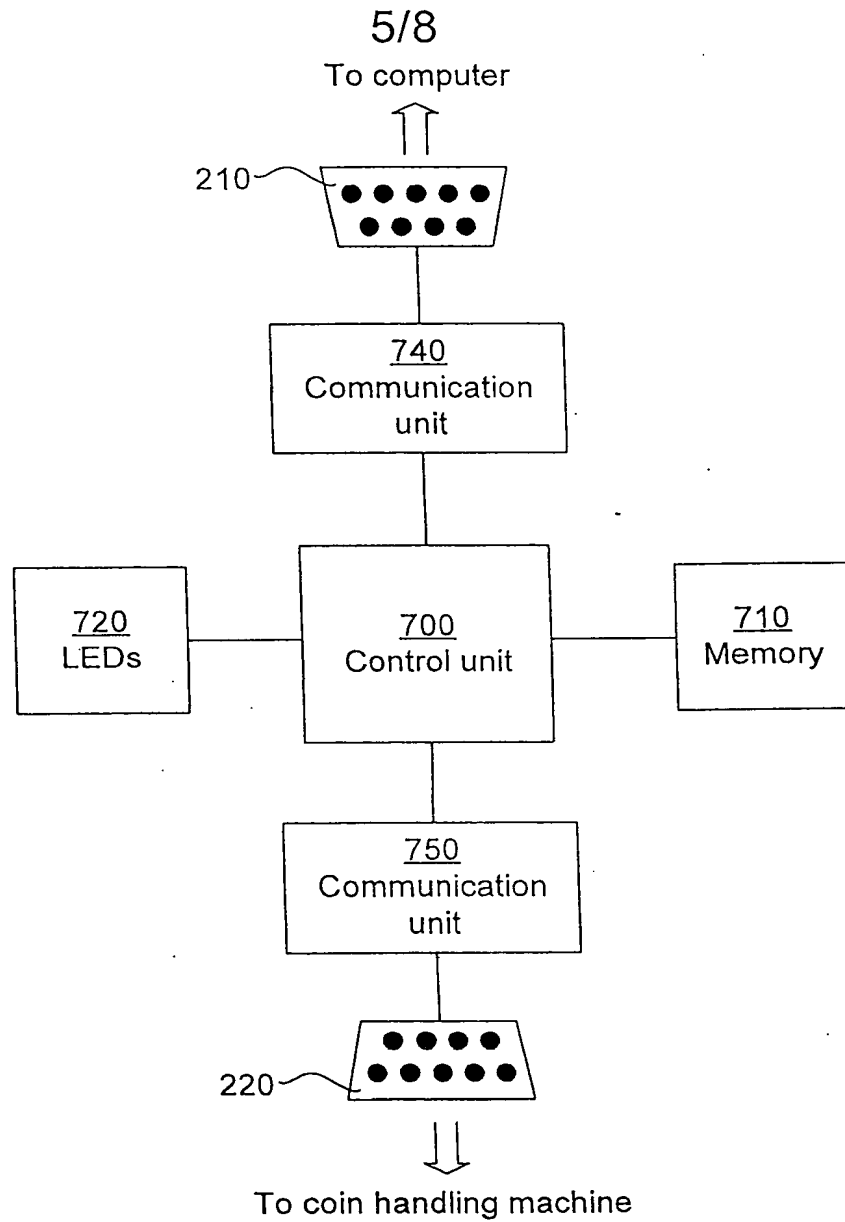
*Fig. 5*

4/8

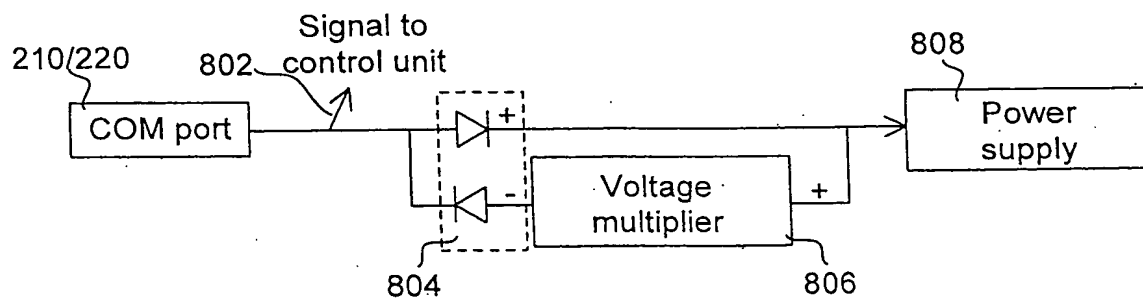


*Fig 6*





*Fig 7*



*Fig 8*

6/8

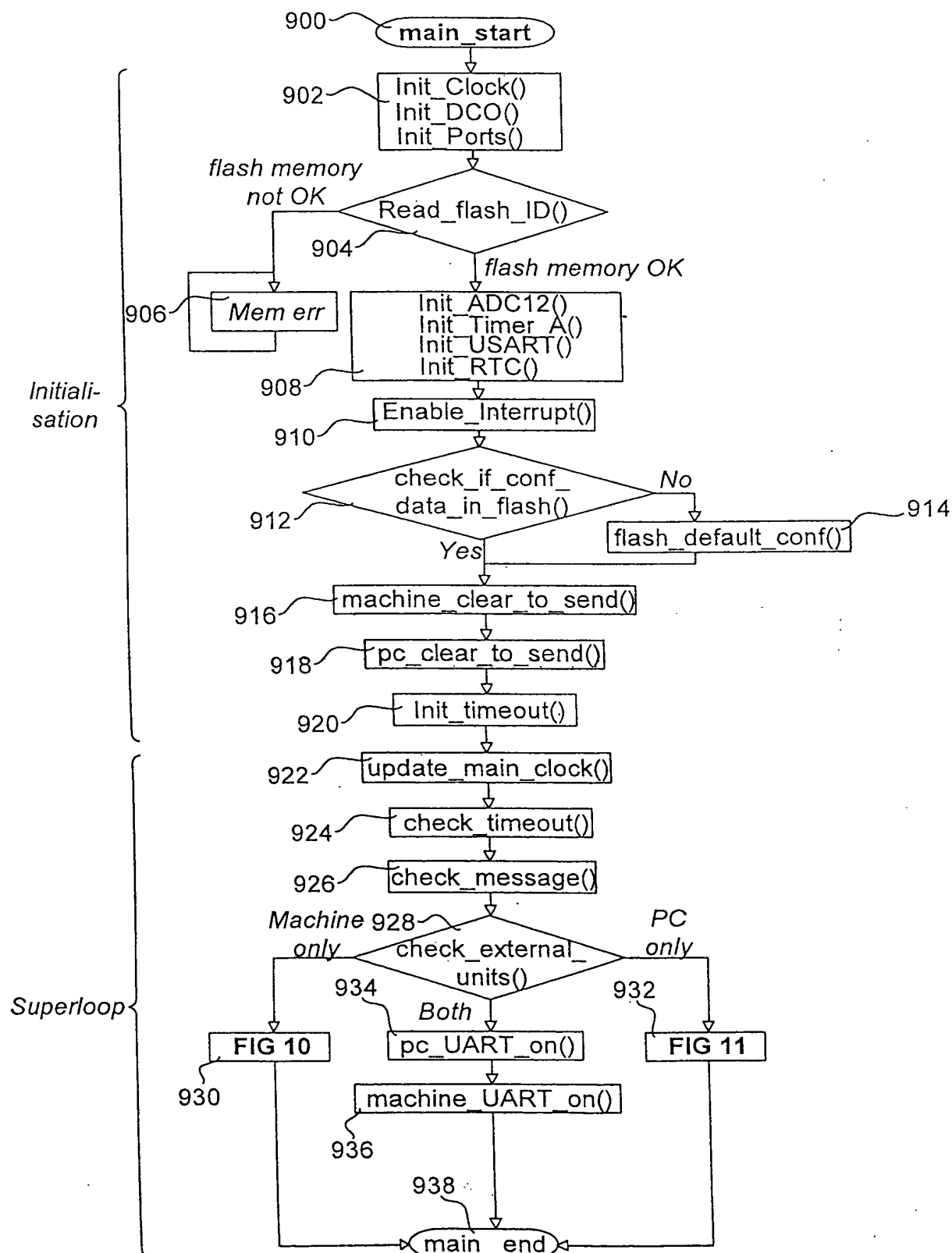


Fig 9

7/8

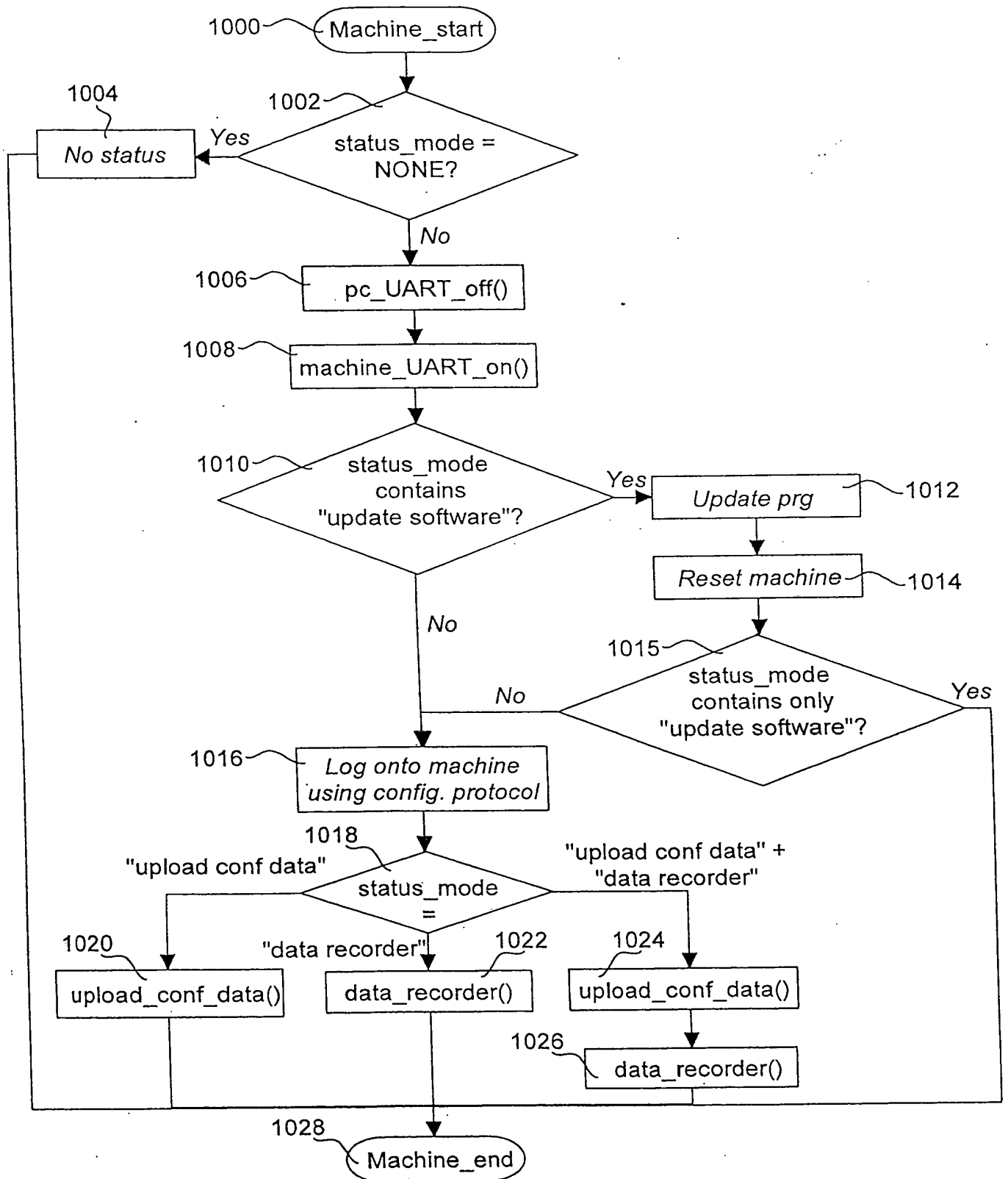
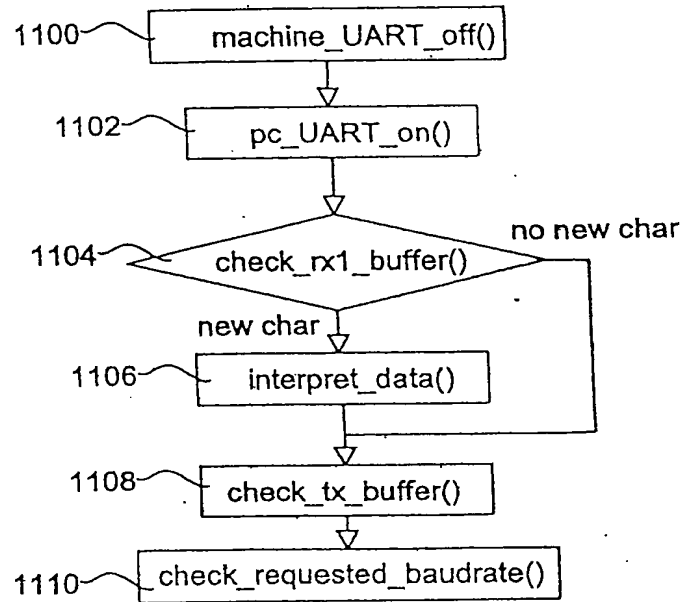


Fig 10

8/8



*Fig 11*

# PRV

PATENT- OCH REGISTRERINGSVERKET  
Patentavdelningen

## Intyg Certificate

Härmed intygas att bifogade kopior överensstämmer med de handlingar som ursprungligen ingivits till Patent- och registreringsverket i nedannämnda ansökan.

*This is to certify that the annexed is a true copy of the documents as originally filed with the Patent- and Registration Office in connection with the following patent application.*



(71) Sökande                      Scan Coin Industries AB, Malmö SE  
Applicant (s)

(21) Patentansökningsnummer    0203131-8  
Patent application number

(86) Ingivningsdatum                      2002-10-21  
Date of filing

Stockholm, 2003-09-22

För Patent- och registreringsverket  
For the Patent- and Registration Office

Hjordis Segerlund

Avgift  
Fee                      170:-

AWAPATENT AB

Kontor/Handläggare

SCAN COIN INDUSTRIES AB

Malmö/Björn Andersson/HAN

Ansökningsnr

Vår referens

SE-2022187

Härmed

PORTABEL ANORDNING SAMT METOD FÖR KONFIGURERING AV  
EN MASKIN SOM HANTERAR MYNT ELLER VÄRDEPAPPER

Tekniskt område

Föreliggande uppfinning avser maskiner för hantering av mynt eller värdepapper och närmare bestämt konfigurering av sådana maskiner.

5

Bakgrundsteknik

Maskiner enligt ovan är givetvis välkända. Som exempel kan nämnas myntsorterare, mynträknare, sedelräknare, sedelläsare, checkläsare, försäljningsautomater, spelautomater, kontantuttagsautomater, maskiner för test av mynt-, sedel- eller checkkvalitet, maskiner för identifiering av falska eller främmande mynt/sedlar/checkar, etc. Sådana maskiner innefattar normalt ett stort antal driftparametrar, inställningar och liknande, vilka gemensamt kan benämnas konfigureringsdata. Även om en given maskin konfigureras på fabrik, kan den av olika skäl behöva omkonfigureras ute på driftstället vid ett eller flera senare tillfälle(n). Omkonfigurering kan även bestå i att programvaran i maskinen behöver uppdateras.

20 Ett exempel på en känd maskin enligt ovan visas i FIG 1 i form av en myntsorterare 100. Myntsorteraren 100 innefattar ett myntintag i anslutning till ett myntuppforderingsverk 106 av typen "hopper", samt vidare en myntdiskriminator 108, ett sorteringsverk 110, en myntretur 112 och ett antal myntmagasin 114. Vidare innefattar maskinen 100 en invändigt belägen mikroprocessor (CPU) eller liknande styrenhet, som även är ansluten till ett minne såsom ett RAM, ROM, EEPROM, flashminne, hårddisk, eller någon kombination därav. Minnet lagrar bl a

25 programvara för maskinen, att exekveras av mikroprocessorn, samt kan även lagra konfigurerings- och arbetsdata för myntdiskriminatoren 108 och sorteringsverket 110.

30

202-74-21

## 2. Produktionskosten

FIG 1 illustrerar ett känt sätt att konfigurera myntsorteraren 100 för att exempelvis ändra dess inställningar och/eller uppdatera dess programvara. En servicetekniker medför en bärbar dator 120 vid sitt besök på det driftställe där maskinen 100 används. Datorn 120 ansluts till maskinen 100 genom exempelvis en seriell kabel 124 mellan ett RS232-gränssnitt på datorn och ett motsvarande RS232-gränssnitt på maskinen. Serviceteknikern kan även behöva ansluta datorn 120 till elnätet via en nätadapter 122 för att inte äventyra servicearbetet, ifall datorns batteri skulle ta slut. När datorn väl har installerats, startar serviceteknikern först dess operativsystem, t ex Microsoft® Windows, och därefter en för ändamålet avsedd konfigureringsapplikation. Med hjälp av denna applikation kan serviceteknikern så konfigurera maskinen 100. Sådan användning av konfigureringsapplikationen erfordrar givetvis att serviceteknikern är välutbildad i densamma.

Som exempel på konfigureringsdata för myntsortera-  
ren 100 kan nämnas hjälptexter och menyer för displayen  
102, eller motsvarande för en på maskinen installerad  
skrivare (ej visad i FIG 1). Andra exempel är parameter-  
värden för myntdiskriminatorn 108 - t ex olika acceptans-  
intervall inom vilka ett antal fysikaliska parametrar  
skall ligga för olika myntvalörer för att dessa skall  
anses vara giltiga. Sådana fysikaliska parametrar kan  
exempelvis vara konduktivitet, permeabilitet, tjocklek,  
diameter eller vikt. Ett möjligt fall, bland många, då  
maskinen 100 kan behöva omkonfigureras är ifall den skall  
användas för en annan valuta än tidigare, eller ifall det  
skett någon förändring i det för maskinen aktuella mynt-  
systemet.

35 Maskinens programvara kan exempelvis behöva uppdateras på grund av att man upptäckt fel, prestanda-

problem eller liknande, eller ifall användaren har beställt ny eller utökad funktionalitet i något avseende.

- Problem och nackdelar med det gängse konfigureringsförfarandet är att det kräver en utbildad service-  
5 tekniker som måste transportera såväl sig själv som en i sammanhanget tung och skrymmande PC (inkl kablage och nätadapter). Vidare tar det tid att koppla in datorn samt starta dess operativsystem och själva applikationen. Dessutom erfordras tillgång till en inte oväsentlig  
10 arbetsyta för uppställning av datorn 120 i nära anslutning till maskinen 100.

#### Sammanfattning av uppfinningen

- Ett syfte med föreliggande uppfinningen är att lösa  
15 eller åtminstone lindra ovanstående problem och nackdelar. Detta syfte har enkelt uttryckt uppnåtts genom att en ny typ av konfigureringshjälpmedel skapats, nämligen en portabel anordning som är avsedd att först anslutas till en dator innehållande nämnda konfigureringsapplikation.  
20 I detta läge emulerar anordningen maskinen för hantering av mynt eller värdepapper, och alla inställningar som användaren av konfigureringsapplikationen gör kommer att mellanlagras i den portabla anordningen. Vid ett senare tillfälle ansluts den portabla anordningen till maskinen  
25 och emulerar i detta läge konfigureringsapplikationen, varvid de i den portabla anordningen mellanlagrade inställningarna kommer att verkställas på maskinen. Den portabla anordningen kan på liknande vis användas för att uppdatera befintlig programvara i maskinen.  
30 Genom detta nya konfigureringshjälpmedel, dvs en portabel anordning enligt uppfinningen, uppnås en rad fördelar. För det första förenklas konfigureringsförfarandet väsentligt, samtidigt som det ställer betydligt lägre krav på utbildningsnivå ute på driftstället. En  
35 välutbildad servicetekniker behöver inte längre själv resa till driftstället utan kan i stället konfigurera den portabla anordningen på hemmaplan. Därefter kan an-



ordningen levereras till driftstället, där en otränad användare kan ansluta den till avsedd maskin, varefter konfigureringen verkställs automatiskt. En fördel i sammanhanget är att den portabla anordningen är lämpad  
5 för miniatyrisering till ett mycket behändigt format. Vidare förkortas servicetiden ute på driftstället, eftersom det inte längre erfordras någon inkoppling och uppstart av konfigureringsdator.

Det ovannämnda syftet uppnås närmare bestämt genom  
10 en portabel anordning och en metod enligt bifogade självständiga patentkrav.

En första aspekt av uppfinningen är således en portabel anordning för konfigurering av en maskin av sådan typ som hanterar mynt eller värdepapper och som  
15 har ett gränssnitt för anslutning av en extern enhet, varigenom maskinen kan konfigureras från den externa enheten. Den portabla anordningen har en styrenhet; ett minne; ett första gränssnitt för anslutning till nämnda externa enhet; samt ett andra gränssnitt för  
20 anslutning till gränssnittet på nämnda maskin; varvid styrenheten är anordnad, då den portabla anordningen är ansluten till den externa enheten via det första gränssnittet, att emulera nämnda maskin samt att från den externa enheten ta emot och i minnet lagra kon-  
25 figureringsdata avsedda för nämnda maskin, och varvid styrenheten är anordnad, då den portabla anordningen är ansluten till nämnda maskin via det andra gränssnittet, att emulera den externa enheten samt att till nämnda maskin avge nämnda i minnet lagrade konfigure-  
30 ringsdata.

I begreppet konfigurering innefattas inställningar av diverse parametrar, referensdata, språkstöd och liknande i maskinen. Vidare innefattas uppdatering av maskinens programvara. Det är emellertid inte nödvändigt  
35 för uppfinningen att bägge dessa moment omfattas av konfigureringsbegreppet. Begreppet konfigureringsdata kan på motsvarande vis innefatta inställningar och/eller para-

5 programvara skall uppdateras.

10 exemplvis enligt RS232-standarden.

15 samt, då den portabla anordningen är ansluten till den

20      åkthet/valör/typ för mynt respektive värdepapper som

25 ning från de första och andra seriella gränssnitten.

35 nämnda externa enhet, till nämnda externa enhet; från

ordningen lagra konfigureringsdata avsedda för nämnda

Ett Patent

2002-10-21

6

Hans-Erik Krasen

maskin; ansluta den portabla anordningen till nämnda maskin; samt till nämnda maskin avge nämnda lagrade konfigureringsdata.

- 5 Andra syften och fördelar med samt särdrag hos uppfinningen framgår av efterföljande detaljerade redogörelse för uppfinningen, av bifogade patentkrav samt av ritningarna.

#### Kort beskrivning av ritningarna

- 10 Uppfinningen kommer nu att beskrivas närmare med hänvisning till medföljande ritningar.

- FIG 1 illustrerar ett tidigare känt sätt att koppla samman en dator med en maskin för hantering av mynt via en seriell kabel i ändamål att konfigurera maskinen, 15 exempelvis för att ändra dess inställningar eller uppdatera programvaran i densamma.

- FIG 2 är en perspektivvy av en portabel anordning enligt uppfinningen för konfigurering av maskinen i FIG 1, valfritt även för inspelning av driftdata från den- 20 samma.

FIG 3 visar den portabla anordningen enligt FIG 2 ansluten till datorn i FIG 1.

FIG 4 visar baksidan på maskinen för hantering av mynt i FIG 1.

- 25 FIG 5 visar den portabla anordningen enligt FIG 2 ansluten till maskinen i FIG 1.

FIG 6 är ett blockschema över centrala komponenter hos den i FIG 1, 4 och 5 visade maskinen för hantering av mynt.

- 30 FIG 7 är ett blockschema över centrala komponenter hos den i FIG 2, 3 och 5 visade portabla anordningen.

FIG 8 är ett blockschema över ett arrangemang för strömförsörjning av komponenterna i FIG 7.

- 35 FIG 9 är ett flödesschema över en huvudrutin för den portabla anordningen enligt FIG 2, 3, 5 och 7.

FIG 10 är ett flödesschema över en rutin för den portabla anordningen, för fallet då den är ansluten till maskinen på det sätt som visas i FIG 5.

FIG 11 är ett flödesschema över en rutin för den portabla anordningen, för fallet då den är ansluten till datorn på det sätt som visas i FIG 3.

## Detaljerad redogörelse för uppfinningen

I FIG 2-5 visas en typisk användning av en portabel anordning 200 enligt en föredragen utföringsform av uppfinningen. Anordningen 200 är som synes utförd som en liten dosa, baserad på ett apparathölje och uppvisande en första kontakt 210 i den ena änden och en andra kontakt 220 i den motsatta änden. Kontakterna 210, 220 är konventionella niopinnars serieportar enligt RS232-standarderna i den föredragna utföringsformen. Den första porten 210 är avsedd för anslutning till en dator 120 (se FIG 3), medan den andra porten 220 är avsedd för anslutning till ett motsvarande gränssnitt 402 på en mynthanteringsmaskin 100 (se FIG 4 och 5). Normalt är inte den portabla anordningen 200 ansluten till både datorn 120 och maskinen 100 samtidigt, utan först datorn (varvid anordningen emulerar maskinen 100 och tar emot konfigureringsdata från datorn 120) och sedan maskinen (varvid anordningen emulerar datorn 120 och vidarebefordrar nämnda konfigureringsdata till maskinen 100). Ett omvänt förfarande, dvs anslutning först till maskinen 100 och sedan till datorn 120, kan emellertid utnyttjas vid användning av anordningen 200 för inspelning av driftdata från maskinen 100 för efterbehandling på datorn 120.

FIG 6 visar ett blockschema över de viktigaste delarna av en mynthanteringsmaskin 600, närmare bestämt myntsorteraren 100 från FIG 1. Maskinen 600 skulle dock lika gärna kunna vara en mynt- eller sedelräknare, en försäljningsautomat, en spelautomat, en kontantuttags-automat (ATM, "Automatic Teller Machine"), en maskin för test av mynt-, sedel- eller checkkvalitet, en maskin för

8.

identifiering av falska eller främmande mynt/sedlar/  
checkar, etc.

En myntmassa som skall sorteras av maskinen 600  
deponeras i ett myntintag 610. Mynten förs via en mynt-  
5 matare 620, exempelvis ett myntuppfördringsverk av typen  
"hopper" (jfr 106 i FIG 1) eller ett transportband, till  
en myntdiskriminator 630 (jfr 108 i FIG 1). Myntdiskri-  
minatorn 630 är operativt ansluten till en styrenhet 632  
i form av en mikroprocessor (CPU) eller liknande, vilken  
10 i sin tur är operativt ansluten till ett minne 634 såsom  
ett RAM, ROM, EEPROM, flashminne, hårddisk, eller någon  
kombination därav. Åtminstone delar av minnet 634 kan  
realiseras av internt minne i styrenheten. Styrenheten  
632 ansvarar för maskinens 600 övergripande arbetssätt,  
15 inbegripande styrning av en display 636 och en knappsats  
638 som bildar ett användargränssnitt (jfr 102 resp 104  
i FIG 1). De konfigureringsdata för maskinen som den por-  
tabla anordningen enligt uppfinningen är avsedd att upp-  
datera finns lagrade i minnet 634. Genom ett konfigure-  
20 ringsgränssnitt 639 (jfr serieporten 402 i FIG 4) kan  
nya/ändrade konfigureringsdata tas emot av styrenheten  
632 från den portabla anordningen 200 samt lagras i  
minnet 634.

Mynthanteringsmaskinen 600 innefattar även en mynt-  
25 retur 640 (jfr 112 i FIG 1), som returnerar ej godkända  
mynt genom en utvändig öppning i maskinen 600, så att  
sådana mynt kan återfås av användaren. Det är myntdiskri-  
minatorn 630 som i sammanhanget avgör om ett mynt skall  
godkännas eller ej genom att detektera diverse fysika-  
30 liska parametrar och därigenom bestämma myntets typ,  
valör, valuta, identitet, äkthet eller liknande. Som  
exempel på fysikaliska parametrar kan nämnas kondukti-  
vit, permeabilitet, tjocklek, diameter eller vikt.  
Efter myntdiskriminatorn 630 följer ett sorteringsverk  
35 650 (jfr 110 i FIG 1), som utnyttjar myntbestämningen  
från myntdiskriminatorn 630 för att sortera in myntet i  
en specifik myntbehållare (jfr 114 i FIG 1) tillhörande

en myntförvaring 660. Myntbehållarna i myntförvaringen 660 är med fördel utvändigt tillgängliga för användaren.

Styrenheten 632 lagrar diverse driftdata i minnet 634. Ett exempel på sådana driftdata är myntstatistik från myntdiskriminatorn 630 - i form av antalet inräknade mynt av respektive valör, antalet avvisade mynt osv, eller mera detaljerat - statistisk för detekterade fysikaliska myntparametrar. Genom konfigureringsgränssnittet 639 kan dessa driftdata - även under pågående drift - avges till den portabla anordningen 200 för "inspelning" av exempelvis myntstatistik. Senare kan denna inspelade myntstatistik laddas ner från den portabla anordningen 200 till datorn 120 för analys eller annan efterbehandling.

Den portabla anordningen 200 visas på blockschemaform i FIG 7. Samtliga i blockschemat ingående komponenter kommer att beskrivas närmare senare men ges först en kort översikt. En styrenhet 700 hanterar anordningens övergripande arbete och styr övriga komponenter. Ett minne 710 är associerat med styrenheten 700 och fungerar som arbets- och programminne för densamma. Minnet 710 används också för att mellanlagra konfigureringsdata som tagits emot från datorn 120 i väntan på att de ska överföras till maskinen 100, respektive mellanlagra driftdata som spelats in från maskinen 100 i väntan på överföring till datorn 120. En uppsättning lysdioder 720 (visas även som 230, 232 i FIG 2) syftar till att signalera olika drift- och feltillstånd till användaren. En första kommunikationsenhet 740 är förbunden med den första serieporten 210, medan en andra kommunikationsenhet 750 är förbunden med den andra serieporten 220.

\* \* \*

Eftersom anordningen 200 emulerar maskinen 100 mot datorn 120 och vice versa sker kommunikationen via kommunikationsenheterna 740 respektive 750 enligt samma generella seriekommunikationsstandard som konventionellt utnyttjas mellan dator och maskin (dvs situation-

10

en i FIG 1). Nedan ges en kort specifikation av denna seriekommunikationsstandard.

För att koppla ihop en maskin med en dator används en RS232-kabel kopplad på följande vis:

5

Dator		Riktning	Maskin	Signal
25	9		9	
stift	stift		stift	
3	2	←	3	Tx
2	3	→	2	Rx
7	5	↔	5	GND
4	7	→	8	CTS (PC ej redo)
5	8	←	4	RTS (Maskin ej redo)

10

När maskinen sätts på startar den med grundinställningar enligt följande: 9600 Baud, ingen paritet, 8 databitar och en stoppbit. Överföringshastigheten ställs sedan normalt om till 115200 Baud.

15

Handskakningssignalerna CTS (Clear To Send) och RTS (Request To Send) används för att få kontroll över flödet. RTS används av maskinen för att indikera när den inte är redo att ta emot data från RX. Skickas något när RTS är aktiv, kan detta resultera i kommunikationsfel. För att undvika detta kan datorn tala om för maskinen att den inte får skicka något. Detta utförs genom att sätta CTS aktiv.

20

Ett mjukvaruprotokoll för konfigurering eller "setup" används för att kommunicera mellan dator och maskin. En allmän benämning på detta protokoll är PCcom. Alla meddelanden som skickas mellan dator och maskin har samma uppbyggnad:

25

Riktning  
PC - Maskin

Data

→

Escape-tecken (ESC 27)

→

Kommando-tecken ('A' - 'Z')

→

Underkommando-tecken ('a' - 'z' eller 'A' - 'Z')

30

↔

Data-bytes

↔

CSUM, enkel byte-checksumma

←

ACK (6) om CSUM är ok

NAK (21) om CSUM inte är ok

+46 40 260316

Pågående arbete

2002-10-11

Förhållande till

11

EM (25) om SUM är OK och det inte finns  
tillräckligt med minne för data.

\* \* \*

5 Med hänvisning åter till FIG 7 samt till FIG 8  
kommer nu den portabla anordningen 200 att beskrivas  
närmare. Därefter beskrivs dess arbetssätt med hänvisning  
till FIG 9-11.

Den portabla anordningen 200 strömförsörjs från  
10 maskinen 100 respektive batorn 120 via utgångarna på  
serieportarna 210, 220, enligt vad som visas i FIG 8.  
Serieportar används normalt enbart till kommunikation,  
varför det är viktigt att inte förstöra kommunikations-  
signalerna. För att skilja kommunikationssignalen 802  
15 från strömförsörjning har dioder 804 arrangerats som  
karaktäristiskt endast släpper igenom spänning i en  
riktning. Genom en spänningsdubblare 806 (t ex National  
Semiconductor LMC7660) kan en positiv spänning erhållas  
ur en negativ spänning för strömförsörjning 808 av anord-  
20 ningens komponenter. Strömförsörjningen 808 kan bl a  
innefatta en spänningsregulator såsom ST Microelectronics  
LE33ABZ för omvandling till lämplig spänningsnivå för  
komponenterna, t ex 3,3 V.

COM-portens spänningsområde ligger enligt RS232-  
25 specifikationen på  $\pm 25V$ . Spänningsområdet är emellertid  
inte lika i alla maskiner. Därför kan det finnas behov av  
att använda skyddsmotstånd på ingångssidan.

Styrenheten 700 kan realiserars av en mikrostyrcrets  
MSP430F149 från Texas Instruments. Denna mikrostyrcrets  
30 har en mycket låg strömförbrukning. Den drar endast  
250  $\mu A$  per MIPS vid 2,2 V matning och är i första hand  
framtagen för tillämpningar med hårda krav på låg energi-  
förbrukning. Matningsområdet ligger mellan 1,8 och 3,6 V.  
Vid en arbetsfrekvens på 5 MHz fås en total strömförbruk-  
35 ning för mikrostyrcretsen på ca 1300  $\mu A$ . Maximal arbets-  
frekvens är 8 MHz vid 3,6 V. Arkitekturen är uppbyggd  
på en 16-bitars RISC-kärna med 27 instruktioner. Varje



12

instruktion tar endast en klockcykel. Instruktionstiden vid 5 MHz blir då 0,2 µs.

Mikrostyret har ett programminne av flashtyp på 32-60kB och ett arbetsminne av RAM-typ på 1-2 KB.

- 5 Dessutom har den en AD-omvandlare som med fördel används till spänningsövervakning och temperaturmätning.

En fördel med mikrostyret MSP430F149 och dess syskon i 14x-serien är den har två USART-kretsar ("Universal Synchronous Asynchronous Receive Transmit").

- 10 I den föredragna utföringsformen realiserades således kommunikationsenheterna 710 och 750 av dessa USART-kretsar och ingår med andra ord också i styrenheten 700.

Givetvis kan styrenheten 700 realiserades av någon annan komponent än den som föreslagna, exempelvis Atmel

- 15 ATMEGA161L 8PI 0143, Atmel TS9C51RD2 eller Microchip PIC18LF452.

Vid realisering av minnet 710 kan man lämpligen beakta parallellprogrammerade flashminnen eller seriellt programmerade EEPROM. Fördelen med EEPROM är att de drar lite ström vid skrivning jämfört med flashminnen. Flashminnen använder sig av "sektorer". Eftersom sektorn i flashminnet måste raderas innan skrivning kan ske, krävs att sektorn hämtas till styrenhetens RAM. RAM-minnet måste därför vara minst lika stort som flashminnets

- 20 sektorstorlek. I den föredragna utföringsformen valdes ett flashminne av typen SST SST28VF040A. Kapaciteten är 512 KB och matningsspänningsområdet 2,7 - 3,6 V. I den föredragna utföringsformen utnyttjas två fysiska minneskapslar, varvid kapaciteten totalt blir 1024 KB. Åtkomsttiden för minnet är 150-200 ns, vilket är kortare än
- 25 åtkomsttiden hos den valda styrenheten. Minnet går att skriva över 100 000 ggr och informationen ska hålla i 100 år. Alternativa val av minne 710 är exempelvis Microchip 24LC515, Microchip 24LC640, ST Microelectronics
- 30 M95256-V eller Atmel AT28C51RD2.

Åtminstone delar av minnet 710 kan realiserades av internt minne i styrenheten, exempelvis programminnet

13

(flash) och arbetsminnet (RAM) i mikrostyrciklet  
MSP430F149.

Den portabla anordningen drive av programvara enligt följande. Programvarans arbetssätt framgår av FIG 9 och är uppbyggd enligt genomströmningsprincipen. Den startar med en initiering för att sedan gå in i en superloop. Här "snurrar" programmet flera tusentals varv per sekund. Omloppshastigheten beror på vad som händer i det aktuella skedet. Superloopen måste vara snabbare än den begränsande faktorn, vilken i detta fall är kommunikationshastigheten. Programmet skulle kunna betraktas som en form av tillståndsmaskin.

Varje "block" läses aldrig upp exekveringen i form av oändlighetsloopar utan återgår hela tiden till superloopen. Ett undantag är om något fel på det externa minnet skulle uppenbaras under initieringen. Det är olämpligt att initieringen i sådana fall fortsätter, eftersom resterande kod är totalt beroende av att det externa minnet fungerar.

Superloopens uppgift är att varje varv övervaka de olika mjukvarublockens tillstånd. Superloopen är hjärnan i mjukvaran och står högst i hierarkin. Den bestämmer bland annat när det är dags att skriva till externminnet eller när det är dags att kommunicera. Till sin hjälp har den mjukvarubufferten, tidsräkningsfunktioner, avbrottsfunktioner, funktionspekare och kommunikationsenheter. Det finns även övervakningsklockor som hela tiden kontrollerar att allt står rätt till.

Mottagningen av information från datorn sker med hjälp av ett avbrotts- Avbrottet fyller en mellanlagringsbuffert som har plats för 20 byte. En cirkulär skrivpekare som ägs av avbrotts- används för att placera data i mellanlagringsbufferten. Cirkulär innebär att den inte alltid startar lagringen från adress noll utan fortsätter lagra från dess nuvarande position. När den når slutet förflyttar den sig till position noll igen. Vid varje nytt datapaket räknas en räknare upp med ett. Denna

14

räknare håller reda på hur mycket som är lagrat i mellanlagringsbufferten och ägs av både avbrottet och superloopen. Räknaren kontrolleras av superloopen varje varv. Då räknaren skiljer sig från noll aktiveras en PCcom-funktion (jfr vad som beskrivits tidigare i detta dokument). PCcom tolkar informationen med hjälp av en läspekare och avgör om den är en del av ett BSC-kommando eller data. Data placeras med hjälp av en annan skrivpekare i en databuffert på 120 byte. Läspekaren är

5 funktion (jfr vad som beskrivits tidigare i detta dokument). PCcom tolkar informationen med hjälp av en läspekare och avgör om den är en del av ett BSC-kommando eller data. Data placeras med hjälp av en annan skrivpekare i en databuffert på 120 byte. Läspekaren är

10 cyklisk och ägs av superloopen. Skrivpekaren till databufferten är däremot inte cyklisk utan startar från position noll vid varje mellanlagringssekvens. Denna pekare ägs av superloopen. Data som nu upplagda i databufferten och är redo att placeras vidare av PCcom. PCcom kan

15 peka direkt på önskad databyte.

För att förenkla programmeringen används två separata mjukvarubuffertar, en för varje avbrott. Mjukvarubuffertarna är tillagda i minsta möjliga storlek utan att kommunikationen försämrats. Anledningen är att de tar upp

20 plats i RAM-minnet som endast är exempelvis 2 KB. Eftersom datorn skickar max 10 byte data per sekvens, är detta en lämplig storlek för databufferten till PCcom.

Mottagning från maskinen 100 sker enligt samma princip som mottagning från datorn 120. Skillnaden är att

25 den portabla anordningen 100 använder en annan mellanlagringsbuffert och annan databuffert. Dessa buffertar har en annan storlek. Mellanlagringsbufferten är på 15 byte och databufferten är på 10 byte.

Sändning till datorn 120 och maskinen 100 sker via en gemensam mjukvarubuffert på 920 byte. Sändningen är

30 inte avbrottsstyrd utan styrs av superloopen. Superloopen håller reda på var data ska skickas med hjälp av flaggor.

Extern minneshantering, dvs hantering av data till och från det externa minnet, sker via funktioner. För att

35 skriva till minnet anropas en skrivefunktion med inparametrarna data och adress. Vid läsning anropas en läsfunktion med inparametern adress, varvid data returneras.

De två minneskapslarna är sammanslagna till ett block med hjälp av mjukvara på låg nivå. Utåt presenteras detta som ett stort sammanhängande minne med linjär adressering från 0x000000 till 0xFFFF, dvs 1 MB.

5       Exempel:

Byte a och b programmeras på adresserna x och y (a och b placeras i olika sektorer). Följande sker:

1) Den sektor som innehåller adress x kopieras från externminnet till minneshanteringsbufferten i mikro-  
10 styrkretsens RAM. Detta sker inte om aktuell sektor redan finns i RAM.

2) Byte a placeras på adress x i minneshanteringsbufferten.

3) Sektor innehållande adress x i externminnet  
15 raderas.

4) Minneshanteringsbufferten programmeras till aktuell sektor i externminnet.

5) Den sektor som innehåller adress y kopieras från externminnet till minneshanteringsbufferten.

20 6) Byte b placeras på adress y i minneshanteringsbufferten.

Byte b programmeras inte till externminnet förrän sektorbyte eller "timeout" sker.

\* \* \*

25 I programflödet enligt FIG 9 sker först initiering (steg 900-920), varefter superloopen inträds (steg 922-938). Under initieringen sker följande:

Programmet startar med att initiera alla globala variabler i steg 902. CPU:s yttre klockfrekvens (X1)  
30 ställs in tills oscillationen stabiliserats. Den inre klockan (DCO) ställs in till 500 kHz. Portarnas riktning ställs in.

Därefter kontrollerar programmet om det externa flashminnet är OK i steg 904. Om minnet inte är OK, indikeras ett felmeddelande i steg 906 med hjälp av lysdioderna 720/230-232. Om minnet däremot är OK, fortsätter programmet med de sista initieringarna i steg 908:

LÄSINSTRUKTION

100-10-21

Röster och Riksdagen

16

AD-omvandlaren stängs in att omvandla 15 ggr per sekund. En timer initieras till att ge avbrott var 100:e ms. Överföringshastigheten ställs in till 9600 Baud. Realtidsklockan (RTC) tilldelas en godtycklig tid och

5 datum.

Därefter sätts det globala avbrottet på i steg 910. I steg 912 kontrollerar programmet om det finns några konfigurationsdata lagrade i det externa minnet. Om så inte är fallet programmeras minnet med en grundkonfiguration i steg 914.

10

I stegen 916 och 918 ges maskin och dator tillstånd att sända. Slutligen stängs timern till RTC i steg 920. Därmed är initieringsdelen avslutad.

I superloopen öppnas först RTC i steg 922. Det

15 kontrolleras i stegen 924 och 926 huruvida eventuell timeout har inträffat respektive vilket meddelande som ska visas. Därefter ser programmet i steg 928 efter vad som är finns anslutet till kommunikationsenheterna. Följande tre fall kan inträffa:

- 20
- Endast datorn är ansluten (steg 932, FIG 11).
  - Endast maskinen är ansluten (steg 930, FIG 10).
  - Både datorn och maskinen är anslutna (steg 934-936).

Då endast datorn är ansluten sker följande enligt

25 FIG 11. Maskinens UART-drivkrets stängs av i steg 1100, och datorns UART-drivkrets sätts på i steg 1102. Det kontrolleras i steg 1104 om något nytt tecken har kommit från datorn. Om så är fallet förs PCcom i steg 1106. Detta steg har till uppgift att tolka inkommande data

30 från datorns konfigureringsprogram på samma sätt som maskinen skulle gjort ifall maskinen vore ansluten direkt till datorn. Efter detta kontrolleras i steg 1108 om något tecken behöver skickas till datorn. Till slut kontrolleras i steg 1110 om datorn begärt ändrad överförings-

35 hastighet.

Då endast maskinen är ansluten sker följande enligt FIG 10. För att förenkla användandet av den portabla

17

anordningen tilldelas olika uppgifter som lagras i "status\_mode". Programmet kontrollerar "status\_mode" och utför given uppgift som beror på aktuell "status\_mode". Det finns totalt åtta olika sätt att kombinera

5 "status\_mode":

Ingen "status\_mode" är vald (steg 1002, 1004): Exekveringen avslutas. I annat fall stängs kommunikationsenheten 740 mot datorn av i steg 1006, medan kommunikationsenheten 750 mot maskinen sätts på i steg 1008.

10 Uppdatera maskinens programvara (steg 1010, 1012): Inloggning på maskinen enligt ett fördefinierat programuppdateringsprotokoll. Efter uppdatering sker mjukvaruomstart av maskinen i steg 1014. Om "status\_mode" indikerar ytterligare något utöver programuppdatering  
15 (kontrolleras i steg 1014), fortsätter exekveringen med inloggning på maskinen enligt ett fördefinierat konfigureringsprotokoll (steg 1016).

Ladda upp konfigureringsdata: Uppdatera konfigureringsdata i maskinen (steg 1018, 1020 resp 1024).

20 Databandsspelare: Spela in driftdata från maskinen och lagra i det externa flashminnet (steg 1018, 1022 resp 1026). Inspelad driftdata kan sedan överföras till datorn för analys eller liknande.

Då både datorn och maskinen är anslutna sätts bägge  
25 kommunikationsenheter 740 och 750 på i steg 934 och 936. I detta läge är den portabla anordningen 200 genomskinlig för datorn och maskinen. Denna uppgift är endast att lagra viktig information samt slicka vidare all inkommande information.

30 Uppfinningen har beskrivits ovan i form av ett par exemplifierande utföringsformer. Dock är uppfinningen inte på något vis begränsad till dessa utan inrymmer många andra varianter, enligt vad som definieras av skyddsomfånget hos de berörda patentkraven och vidare  
35 enkelt inses av fackmannen. En möjlighet i sammanhanget är att utnyttja en och samma fysiska kommunikationsport och/eller kommunikationsenhet hos den portabla anordning-

18

en för såväl anslutningen mot datorn som för anslutningen  
mot maskinen. I en sådan utföringsform får styrenheten i  
den portabla anordningen arrangeras att känna igen vilken  
av datorn eller maskinen som för närvarande är ansluten  
5 och agera i beroende därav.

9  
7  
6  
5  
4  
3  
2  
1

## BESKREIVNING

1. Portabel anordning (200) för konfigurering av  
5 en maskin (100) av sådan typ som hanterar mynt eller  
värdepapper och som har ett gränssnitt (402) för an-  
slutning av en extern enhet (120). Varigenom maskinen  
kan konfigureras från den externa enheten, varvid den  
portabla anordningen kännetecknas av
- 10 en styrenhet (700);  
ett minne (710);  
ett första gränssnitt (210, 740) för anslutning  
till nämnda externa enhet (120); samt  
ett andra gränssnitt (220, 750) för anslutning  
15 till gränssnittet (402) på nämnda maskin (100);  
varvid styrenheten är anordnad, då den portabla  
anordningen är ansluten till den externa enheten via  
det första gränssnittet, att emulera nämnda maskin  
samt att från den externa enheten ta emot och i minnet  
20 lagra konfigureringsdata avsedda för nämnda maskin,  
och  
varvid styrenheten är anordnad, då den portabla  
anordningen är ansluten till nämnda maskin via det  
andra gränssnittet, att emulera den externa enheten  
25 samt att till nämnda maskin avge nämnda i minnet  
lagrade konfigureringsdata.
2. Portabel anordning enligt krav 1, varvid den  
externa enheten (120) är en dator av typen bärbar dator,  
30 persondator, handdator, arbetsstation eller liknande.
3. Portabel anordning enligt krav 1 eller 2, varvid  
de första och andra gränssnitten (210, 740, 220, 750) är  
seriella.
- 35 4. Portabel anordning enligt något av ovanstående  
krav, varvid nämnda konfigureringsdata innefattar åtmin-



20

stone något av följande: inställningar och/eller parametrar för giltighet/åktighet/valör/typ för mynt respektive värdepapper; referensdata eller data avseende språkstöd.

5 5. Portabel anordning enligt något av ovanstående krav, varvid nämnda konfigureringsdata innefattar programvara för exekvering i en maskinen (100) ingående mikroprocessor (632).

10 6. Portabel anordning enligt något av ovanstående krav,

varvid styrenheten (700) är anordnad, då den portabla anordningen (200) är ansluten till nämnda maskin (100) via det andra gränssnittet (210, 750),  
15 att ta emot driftdata från nämnda maskin och lagra dessa i minnet (710); samt

varvid styrenheten (700) är anordnad, då den portabla anordningen (200) är ansluten till den externa enheten (120) via det första gränssnittet  
20 (210, 740), att till den externa enheten överföra nämnda i minnet lagrade driftdata.

7. Portabel anordning enligt krav 6, varvid nämnda driftdata innefattar antal/giltighet/åktighet/  
25 valör/typ för mynt respektive värdepapper som hanteras av maskinen (100) och/eller av maskinen detekterade fysikaliska parametrar för sådana mynt respektive värdepapper.

30 8. Portabel anordning enligt krav 3, varvid anordningen erhåller strömförsörjning från de första och andra seriella gränssnitten (210, 740, 220, 750).

9. Portabel anordning enligt något av ovanstående  
35 krav, varvid det första och andra gränssnittet utgörs av en och samma fysiska enhet.

21

10. Portabel anordning enligt något av ovanstående krav, varvid nämnda maskin (100) är en myntsorterare eller mynträknare.

5 11. Metod för konfigurering av en maskin (100) av sådan typ som hanterar mynt eller värdepapper och som kan konfigureras från en extern enhet (120), varvid metoden kännetecknas av stegen att

10 ansluta en portabel anordning (200), annan än nämnda maskin (100) och nämnda externa enhet (120), till nämnda externa enhet (120);

från den externa enheten ta emot och i den portabla anordningen lagra konfigureringsdata avsedda för nämnda maskin;

15 ansluta den portabla anordningen till nämnda maskin; samt

till nämnda maskin avge nämnda lagrade konfigureringsdata.

20 12. Metod enligt krav 11, varvid den externa enheten (120) är en dator av någon bärbar dator, persondator, handdator, arbetsstation eller liknande.

25 13. Metod enligt krav 11 eller 12, varvid nämnda konfigureringsdata innefattar åtminstone något av följande: inställningar och/eller parametrar för giltighet/äkthet/valör/typ för mynt respektive värdepapper; referensdata; eller data förseende språkstöd.

30 14. Metod enligt något av kraven 11-13, varvid nämnda konfigureringsdata innefattar programvara för exekvering i nämnda maskin (100).

35 15. Metod enligt något av kraven 11-14, vidare innefattande stegen att ansluta den portabla anordningen (200) till nämnda maskin (100);

22

ansluta den portabla anordningen till den externa enheten (120); samt

16. Metod enligt krav 15, varvid nämnda drift-  
data innefattar antal/gastighet/åkthet/valör/typ för  
10 mynt respektive värdepapper som hanterats av maskinen  
(100), och/eller av maskinen detekterade fysikaliska  
parametrar för sådana mynt respektive värdepapper.

17. Metod enligt något av kraven 11-16, varvid  
15 nämnda maskin (100) är en myntsorterare eller mynt-  
räknare.

## SAMMADRAG

En portabel anordning (200) presenteras för  
5 konfigurerings av en maskin (100) av sådan typ som  
hanterar mynt eller värdepapper och som har ett gränssnitt (402) för anslutning av en extern enhet (120),  
varigenom maskinen kan konfigureras från den externa  
enheten. Den portabla anordningen har en styrenhet  
10 (700); ett minne (710); ett första gränssnitt (210,  
740) för anslutning till nämnda externa enhet (120);  
samt ett andra gränssnitt (220, 750) för anslutning  
till gränssnittet (402) på nämnda maskin (100).  
Styrenheten är anordnad, då den portabla anordningen  
15 är ansluten till den externa enheten via det första  
gränssnittet, att emulera nämnda maskin samt att från  
den externa enheten ta emot och i minnet lagra kon-  
figureringsdata avsedda för nämnda maskin. Styrenheten  
är vidare anordnad, då den portabla anordningen är an-  
20 sluten till nämnda maskin via det andra gränssnittet,  
att emulera den externa enheten samt att till nämnda  
maskin avge nämnda i minnet lagrade konfigurerings-  
data.

25 Att publiceras tillsammans med FIG 7.

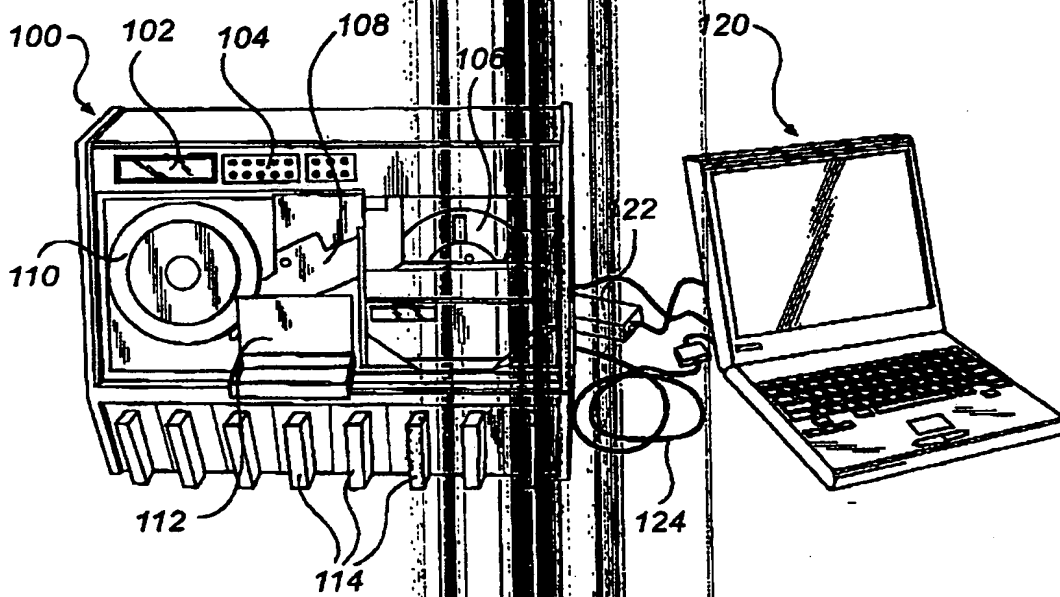
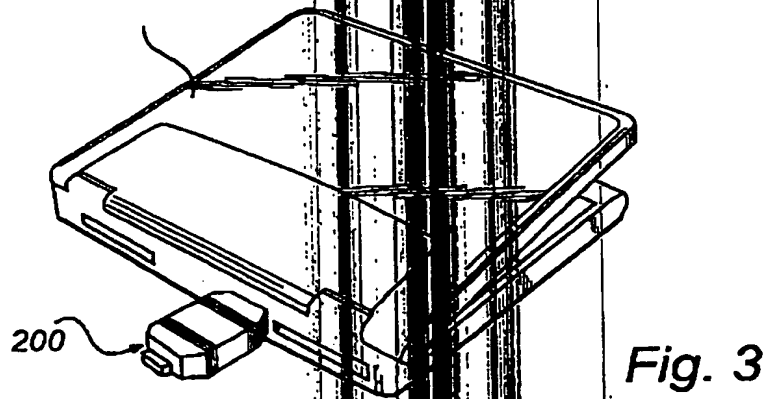
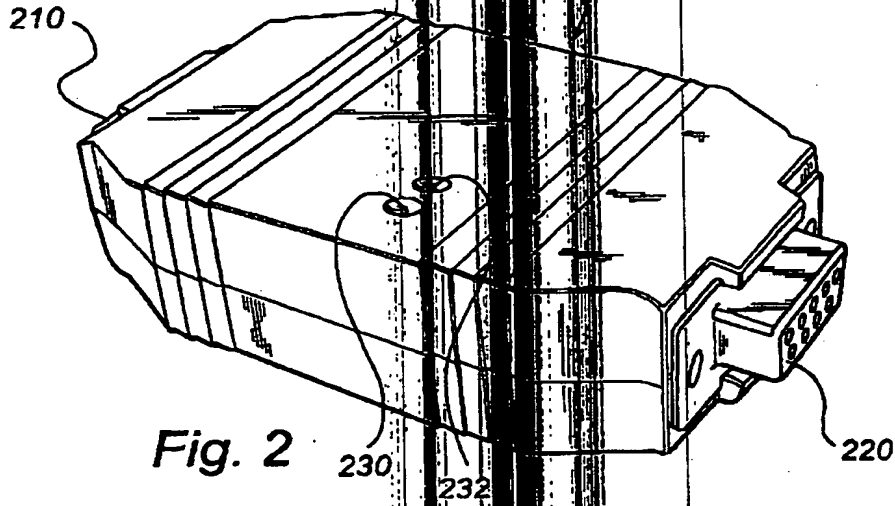
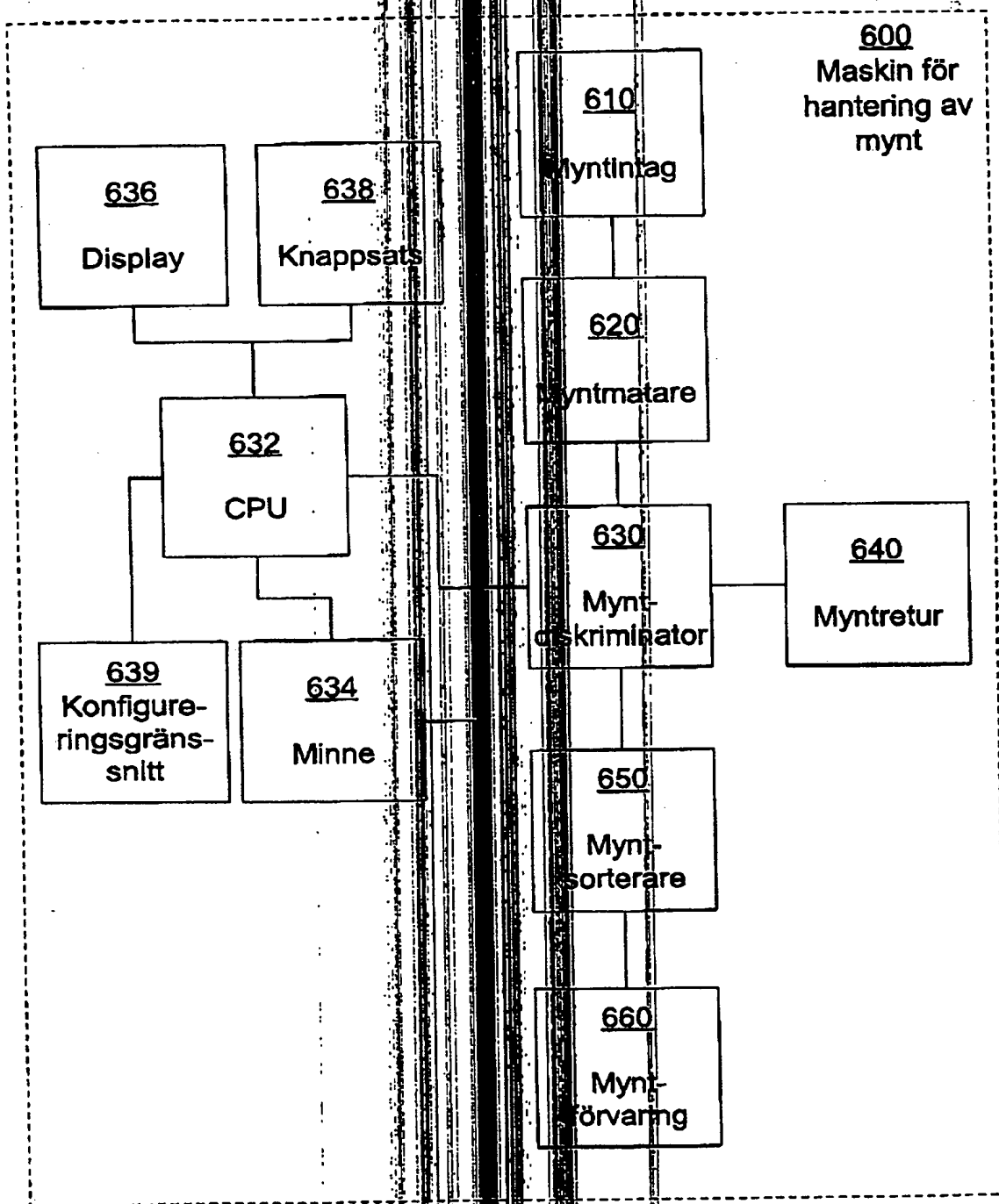


Fig. 1



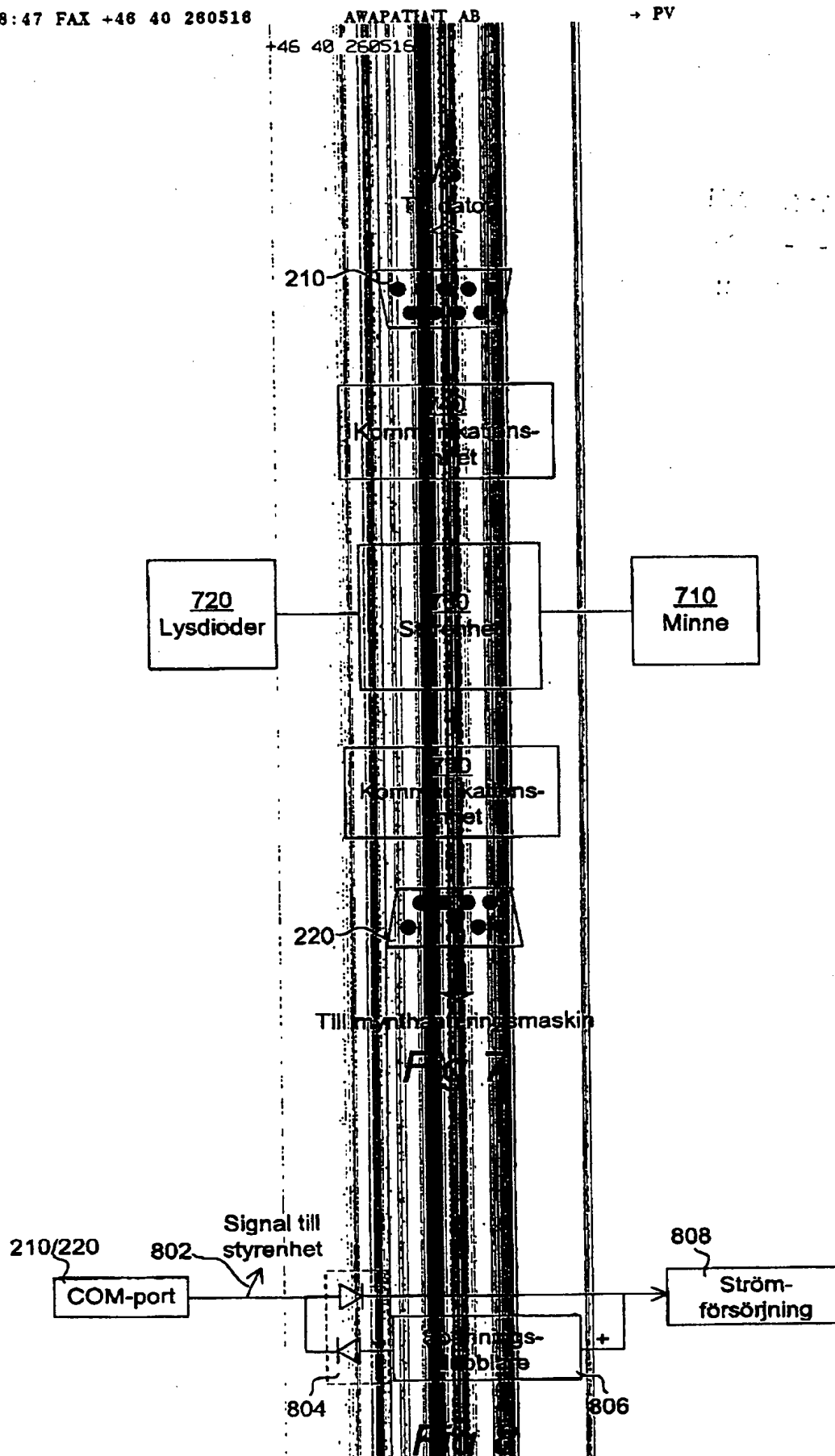


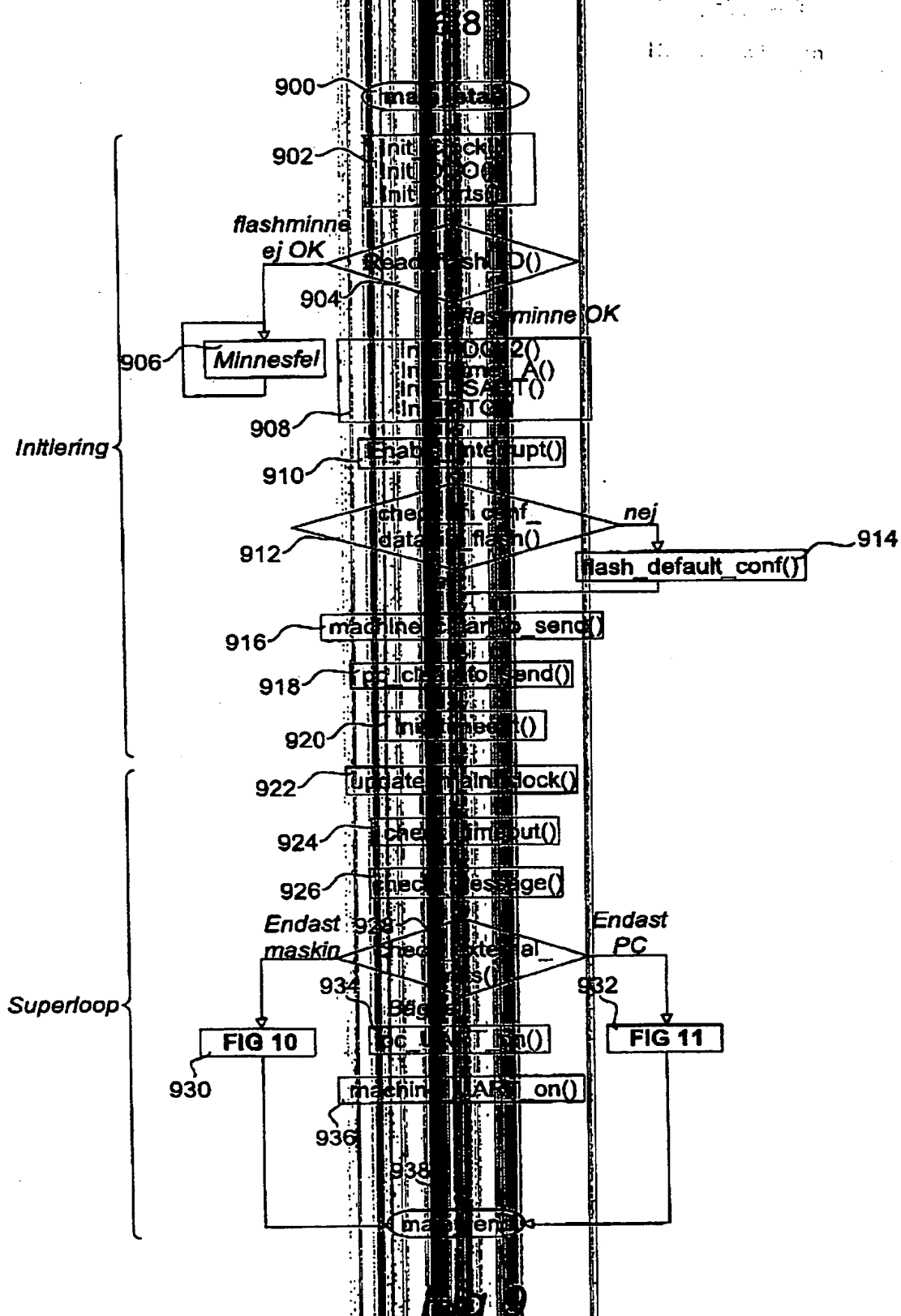
48



FM16







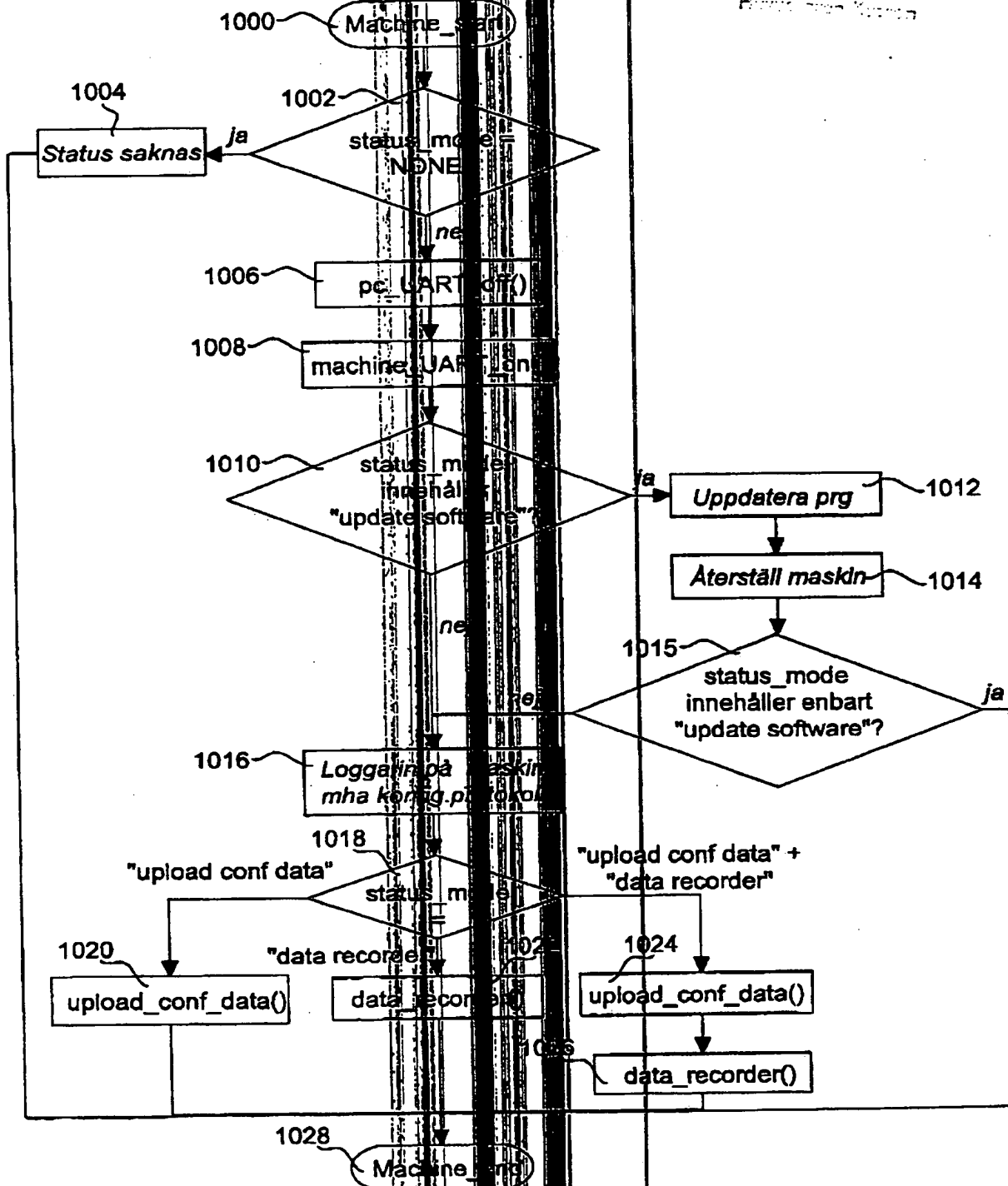


Fig 10

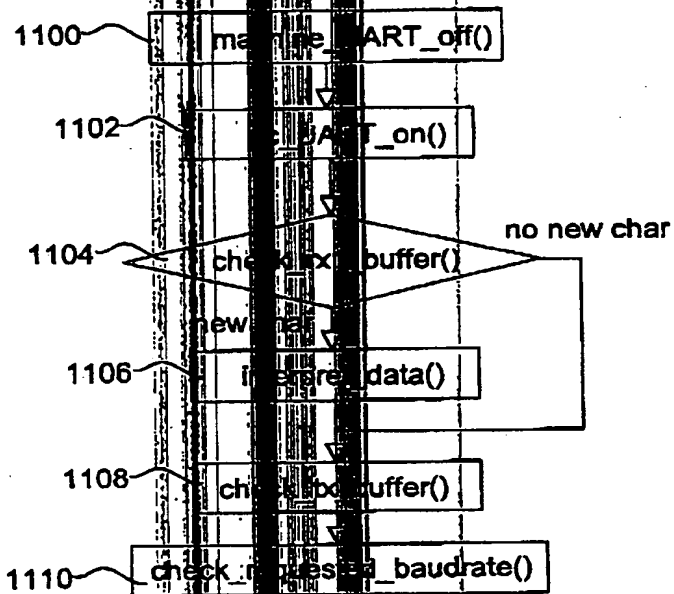


Fig 11

PORTABEL ANORDNING SAMT METOD FÖR KONFIGURERING AV  
EN MASKIN SOM HANTERAR MYNT ELLER VÄRDEPAPPER

Tekniskt område

Föreliggande uppfinning avser maskiner för hantering av mynt eller värdepapper och närmare bestämt konfigureringsdata av sådana maskiner.

5

Bakgrundsteknik

Maskiner enligt ovan är givetvis välkända. Som exempel kan nämnas myntsorterare, mynträknare, sedelräknare, sedelläsare, checkläsare, försäljningsautomater, spelautomater, kontantuttagsautomater, maskiner för test av mynt-, sedel- eller checkkvalitet, maskiner för identifiering av falska eller främmande mynt/sedlar/checkar, etc. Sådana maskiner innefattar normalt ett stort antal driftparametrar, inställningar och liknande, vilka gemensamt kan benämnas konfigureringsdata. Även om en given maskin konfigureras på fabrik, kan den av olika skäl behöva omkonfigureras ute på driftstället vid ett eller flera senare tillfälle(n). Omkonfigureringsdata kan även bestå i att programvaran i maskinen behöver uppdateras.

15

Ett exempel på en känd maskin enligt ovan visas i FIG 1 i form av en myntsorterare 100. Myntsorteraren 100 innefattar ett myntintag i anslutning till ett myntuppforderingsverk 106 av typen "hopper", samt vidare en myntdiskriminator 108, ett sorteringsverk 110, en myntretur 112 och ett antal myntmagasin 114. Vidare innefattar maskinen 100 en invändigt belägen mikroprocessor (CPU) eller liknande styrenhet, som även är ansluten till ett minne såsom ett RAM, ROM, EEPROM, flashminne, hårddisk, eller någon kombination därav. Minnet lagrar bl a programvara för maskinen, att exekveras av mikroprocessorn, samt kan även lagra konfigurerings- och arbetsdata för myntdiskriminators 108 och sorteringsverket 110.

25

30

En av uppgifterna för programvaran är att bilda ett användargränssnitt tillsammans med en frontmonterad display 102 och knappsats 104.

FIG 1 illustrerar ett känt sätt att konfigurera myntsorteraren 100 för att exempelvis ändra dess inställningar och/eller uppdatera dess programvara. En servicetekniker medför en bärbar dator 120 vid sitt besök på det driftställe där maskinen 100 används. Datorn 120 ansluts till maskinen 100 genom exempelvis en seriell kabel 124 mellan ett RS232-gränssnitt på datorn och ett motsvarande RS232-gränssnitt på maskinen. Serviceteknikern kan även behöva ansluta datorn 120 till elnätet via en nätadapter 122 för att inte äventyra servicearbetet, ifall datorns batteri skulle ta slut. När datorn väl har installerats, startar serviceteknikern först dess operativsystem, t ex Microsoft® Windows, och därefter en för ändamålet avsedd konfigureringsapplikation. Med hjälp av denna applikation kan serviceteknikern så konfigurera maskinen 100. Sådan användning av konfigureringsapplikationen erfordrar givetvis att serviceteknikern är välutbildad i densamma.

Som exempel på konfigureringsdata för myntsorteraren 100 kan nämnas hjälptexter och menyer för displayen 102, eller motsvarande för en på maskinen installerad skrivare (ej visad i FIG 1). Andra exempel är parametervärden för myntdiskriminatoren 108 - t ex olika acceptansintervall inom vilka ett antal fysikaliska parametrar skall ligga för olika myntvalörer för att dessa skall anses vara giltiga. Sådana fysikaliska parametrar kan exempelvis vara konduktivitet, permeabilitet, tjocklek, diameter eller vikt. Ett möjligt fall, bland många, då maskinen 100 kan behöva omkonfigureras är ifall den skall användas för en annan valuta än tidigare, eller ifall det skett någon förändring i det för maskinen aktuella myntsystemet.

Maskinens programvara kan exempelvis behöva uppdateras på grund av att man upptäckt fel, prestanda-

problem eller liknande, eller ifall användaren har beställt ny eller utökad funktionalitet i något avseende.

Problem och nackdelar med det gängse konfigureringsförfarandet är att det kräver en utbildad service-  
 5 tekniker som måste transportera såväl sig själv som en i sammanhanget tung och skrymmande PC (inkl kablage och nätadapter). Vidare tar det tid att koppla in datorn samt starta dess operativsystem och själva applikationen. Dessutom erfordras tillgång till en inte oväsentlig  
 10 arbetsyta för uppställning av datorn 120 i nära anslutning till maskinen 100.

#### Sammanfattning av uppfinningen

Ett syfte med föreliggande uppfinningen är att lösa  
 15 eller åtminstone lindra ovanstående problem och nackdelar. Detta syfte har enkelt uttryckt uppnåtts genom att en ny typ av konfigureringshjälpmedel skapats, nämligen en portabel anordning som är avsedd att först anslutas till en dator innehållande nämnda konfigureringsapplikation.  
 20 I detta läge emulerar anordningen maskinen för hantering av mynt eller värdepapper, och alla inställningar som användaren av konfigureringsapplikationen gör kommer att mellanlagras i den portabla anordningen. Vid ett senare tillfälle ansluts den portabla anordningen till maskinen  
 25 och emulerar i detta läge konfigureringsapplikationen, varvid de i den portabla anordningen mellanlagrade inställningarna kommer att verkställas på maskinen. Den portabla anordningen kan på liknande vis användas för att uppdatera befintlig programvara i maskinen.

30 Genom detta nya konfigureringshjälpmedel, dvs en portabel anordning enligt uppfinningen, uppnås en rad fördelar. För det första förenklas konfigureringsförfarandet väsentligt, samtidigt som det ställer betydligt lägre krav på utbildningsnivå ute på driftstället. En  
 35 välutbildad servicetekniker behöver inte längre själv resa till driftstället utan kan i stället konfigurera den portabla anordningen på hemmaplan. Därefter kan an-

ordningen levereras till driftstället, där en otränad användare kan ansluta den till avsedd maskin, varefter konfigureringen verkställs automatiskt. En fördel i sammanhanget är att den portabla anordningen är lämpad  
 5 för miniatyrisering till ett mycket behändigt format. Vidare förkortas servicetiden ute på driftstället, eftersom det inte längre erfordras någon inkoppling och uppstart av konfigureringsdator.

Det ovannämnda syftet uppnås närmare bestämt genom  
 10 en portabel anordning och en metod enligt bifogade självständiga patentkrav.

En första aspekt av uppfinningen är således en portabel anordning för konfigurering av en maskin av sådan typ som hanterar mynt eller värdepapper och som  
 15 har ett gränssnitt för anslutning av en extern enhet, varigenom maskinen kan konfigureras från den externa enheten. Den portabla anordningen har en styrenhet; ett minne; ett första gränssnitt för anslutning till nämnda externa enhet; samt ett andra gränssnitt för  
 20 anslutning till gränssnittet på nämnda maskin; varvid styrenheten är anordnad, då den portabla anordningen är ansluten till den externa enheten via det första gränssnittet, att emulera nämnda maskin samt att från den externa enheten ta emot och i minnet lagra kon-  
 25 figureringsdata avsedda för nämnda maskin, och varvid styrenheten är anordnad, då den portabla anordningen är ansluten till nämnda maskin via det andra gränssnittet, att emulera den externa enheten samt att till nämnda maskin avge nämnda i minnet lagrade konfigure-  
 30 ringsdata.

I begreppet konfigurering innefattas inställningar av diverse parametrar, referensdata, språkstöd och liknande i maskinen. Vidare innefattas uppdatering av maskinens programvara. Det är emellertid inte nödvändigt  
 35 för uppfinningen att bägge dessa moment omfattas av konfigureringsbegreppet. Begreppet konfigureringsdata kan på motsvarande vis innefatta inställningar och/eller para-



metrar för giltighet/äkthet/valör/typ för mynt respektive värdepapper; referensdata; eller data avseende språkstöd. Vidare kan begreppet konfigureringsdata innefatta programkod i någon maskinläsbar form, med vilken maskinens programvara skall uppdateras.

Den externa enheten kan vara en dator av typen bärbar dator, persondator, handdator, arbetsstation eller liknande.

De första och andra gränssnitten kan vara seriella, exempelvis enligt RS232-standard.

Enligt en utföringsform är styrenheten vidare anordnad, då den portabla anordningen är ansluten till nämnda maskin via det andra gränssnittet, att ta emot driftdata från nämnda maskin och lagra dessa i minnet; samt, då den portabla anordningen är ansluten till den externa enheten via det första gränssnittet, att till den externa enheten överföra nämnda i minnet lagrade driftdata.

Nämnda driftdata kan innefatta antal/giltighet/äkthet/valör/typ för mynt respektive värdepapper som hanterats av maskinen, och/eller av maskinen detekterade fysikaliska parametrar för sådana mynt respektive värdepapper.

Den portabla anordningen kan erhålla strömförsörjning från de första och andra seriella gränssnitten.

Enligt en alternativ utföringsform utgörs det första och andra gränssnittet av en och samma fysiska enhet.

Maskinen enligt ovan kan med fördel vara en myntsorterare eller mynträknare.

En andra aspekt av uppfinningen är en metod för konfigureringsdata av en maskin av sådan typ som hanterar mynt eller värdepapper och som kan konfigureras från en extern enhet. Metoden innefattar stegen att ansluta en portabel anordning, annan än nämnda maskin och nämnda externa enhet, till nämnda externa enhet; från den externa enheten ta emot och i den portabla anordningen lagra konfigureringsdata avsedda för nämnda

maskin; ansluta den portabla anordningen till nämnda maskin; samt till nämnda maskin avge nämnda lagrade konfigureringsdata.

Andra syften och fördelar med samt särdrag hos  
5 uppfinningen framgår av efterföljande detaljerade redogörelse för uppfinningen, av bifogade patentkrav samt av ritningarna.

#### Kort beskrivning av ritningarna

10 Uppfinningen kommer nu att beskrivas närmare med hänvisning till medföljande ritningar.

FIG 1 illustrerar ett tidigare känt sätt att koppla samman en dator med en maskin för hantering av mynt via en seriell kabel i ändamål att konfigurera maskinen,  
15 exempelvis för att ändra dess inställningar eller uppdatera programvaran i densamma.

FIG 2 är en perspektivvy av en portabel anordning enligt uppfinningen för konfigurering av maskinen i FIG 1, valfritt även för inspelning av driftdata från den-  
20 samma.

FIG 3 visar den portabla anordningen enligt FIG 2 ansluten till datorn i FIG 1.

FIG 4 visar baksidan på maskinen för hantering av mynt i FIG 1.

25 FIG 5 visar den portabla anordningen enligt FIG 2 ansluten till maskinen i FIG 1.

FIG 6 är ett blockschema över centrala komponenter hos den i FIG 1, 4 och 5 visade maskinen för hantering av mynt.

30 FIG 7 är ett blockschema över centrala komponenter hos den i FIG 2, 3 och 5 visade portabla anordningen.

FIG 8 är ett blockschema över ett arrangemang för strömförsörjning av komponenterna i FIG 7.

35 FIG 9 är ett flödesschema över en huvudrutin för den portabla anordningen enligt FIG 2, 3, 5 och 7.

FIG 10 är ett flödesschema över en rutin för den portabla anordningen, för fallet då den är ansluten till maskinen på det sätt som visas i FIG 5.

FIG 11 är ett flödesschema över en rutin för den portabla anordningen, för fallet då den är ansluten till datorn på det sätt som visas i FIG 3.

#### Detaljerad redogörelse för uppfinningen

I FIG 2-5 visas en typisk användning av en portabel anordning 200 enligt en föredragen utföringsform av uppfinningen. Anordningen 200 är som synes utförd som en liten dosa, baserad på ett apparathölje och uppvisande en första kontakt 210 i den ena änden och en andra kontakt 220 i den motsatta änden. Kontakterna 210, 220 är konventionella niopinnars serieportar enligt RS232-standarden i den föredragna utföringsformen. Den första porten 210 är avsedd för anslutning till en dator 120 (se FIG 3), medan den andra porten 220 är avsedd för anslutning till ett motsvarande gränssnitt 402 på en mynthanteringsmaskin 100 (se FIG 4 och 5). Normalt är inte den portabla anordningen 200 ansluten till både datorn 120 och maskinen 100 samtidigt, utan först datorn (varvid anordningen emulerar maskinen 100 och tar emot konfigureringsdata från datorn 120) och sedan maskinen (varvid anordningen emulerar datorn 120 och vidarebefordrar nämnda konfigureringsdata till maskinen 100). Ett omvänt förfarande, dvs anslutning först till maskinen 100 och sedan till datorn 120, kan emellertid utnyttjas vid användning av anordningen 200 för inspelning av driftdata från maskinen 100 för efterbehandling på datorn 120.

FIG 6 visar ett blockschema över de viktigaste delarna av en mynthanteringsmaskin 600, närmare bestämt myntsorteraren 100 från FIG 1. Maskinen 600 skulle dock lika gärna kunna vara en mynt- eller sedelräknare, en försäljningsautomat, en spelautomat, en kontantuttagsautomat (ATM, "Automatic Teller Machine"), en maskin för test av mynt-, sedel- eller checkkvalitet, en maskin för

identifiering av falska eller främmande mynt/sedlar/  
checkar, etc.

En myntmassa som skall sorteras av maskinen 600  
deponeras i ett myntintag 610. Mynten förs via en mynt-  
5 matare 620, exempelvis ett myntuppforderingsverk av typen  
"hopper" (jfr 106 I FIG 1) eller ett transportband, till  
en myntdiskriminator 630 (jfr 108 I FIG 1). Myntdiskri-  
minatorn 630 är operativt ansluten till en styrenhet 632  
i form av en mikroprocessor (CPU) eller liknande, vilken  
10 i sin tur är operativt ansluten till ett minne 634 såsom  
ett RAM, ROM, EEPROM, flashminne, hårddisk, eller någon  
kombination därav. Åtminstone delar av minnet 634 kan  
realiseras av internt minne i styrenheten. Styrenheten  
632 ansvarar för maskinens 600 övergripande arbetssätt,  
15 inbegripande styrning av en display 636 och en knappsats  
638 som bildar ett användargränssnitt (jfr 102 resp 104  
i FIG 1). De konfigureringsdata för maskinen som den por-  
tabla anordningen enligt uppfinningen är avsedd att upp-  
datera finns lagrade i minnet 634. Genom ett konfigure-  
20 ringsgränssnitt 639 (jfr serieporten 402 i FIG 4) kan  
nya/ändrade konfigureringsdata tas emot av styrenheten  
632 från den portabla anordningen 200 samt lagras i  
minnet 634.

Mynthanteringsmaskinen 600 innefattar även en mynt-  
25 retur 640 (jfr 112 i FIG 1), som returnerar ej godkända  
mynt genom en utvändig öppning i maskinen 600, så att  
sådana mynt kan återfås av användaren. Det är myntdiskri-  
minatorn 630 som i sammanhanget avgör om ett mynt skall  
godkännas eller ej genom att detektera diverse fysika-  
30 liska parametrar och därigenom bestämma myntets typ,  
valör, valuta, identitet, äkthet eller liknande. Som  
exempel på fysikaliska parametrar kan nämnas kondukti-  
vit, permeabilitet, tjocklek, diameter eller vikt.  
Efter myntdiskriminatorn 630 följer ett sorteringsverk  
35 650 (jfr 110 i FIG 1), som utnyttjar myntbestämningen  
från myntdiskriminatorn 630 för att sortera in myntet i  
en specifik myntbehållare (jfr 114 i FIG 1) tillhörande

en myntförvaring 660. Myntbehållarna i myntförvaringen 660 är med fördel utvändigt tillgängliga för användaren.

Styrenheten 632 lagrar diverse driftdata i minnet 634. Ett exempel på sådana driftdata är myntstatistik från myntdiskriminators 630 - i form av antalet inräknade mynt av respektive valör, antalet avvisade mynt osv, eller mera detaljerat - statistisk för detekterade fysikaliska myntparametrar. Genom konfigureringsgränssnittet 639 kan dessa driftdata - även under pågående drift - avges till den portabla anordningen 200 för "inspelning" av exempelvis myntstatistik. Senare kan denna inspelade myntstatistik laddas ner från den portabla anordningen 200 till datorn 120 för analys eller annan efterbehandling.

Den portabla anordningen 200 visas på blockschemaform i FIG 7. Samtliga i blockschemat ingående komponenter kommer att beskrivas närmare senare men ges först en kort översikt. En styrenhet 700 hanterar anordningens övergripande arbete och styr övriga komponenter. Ett minne 710 är associerat med styrenheten 700 och fungerar som arbets- och programminne för densamma. Minnet 710 används också för att mellanlagra konfigureringsdata som tagits emot från datorn 120 i väntan på att de ska överföras till maskinen 100, respektive mellanlagra driftdata som spelats in från maskinen 100 i väntan på överföring till datorn 120. En uppsättning lysdioder 720 (visas även som 230, 232 i FIG 2) syftar till att signalera olika drift- och feltillstånd till användaren. En första kommunikationsenhet 740 är förbunden med den första serieporten 210, medan en andra kommunikationsenhet 750 är förbunden med den andra serieporten 220.

\* \* \*

Eftersom anordningen 200 emulerar maskinen 100 mot datorn 120 och vice versa sker kommunikationen via kommunikationsenheterna 740 respektive 750 enligt samma generella seriekommunikationsstandard som konventionellt utnyttjas mellan dator och maskin (dvs situation-

en i FIG 1). Nedan ges en kort specifikation av denna seriekommunikationsstandard.

För att koppla ihop en maskin med en dator används en RS232-kabel kopplad på följande vis:

5

Dator		Riktning	Maskin	Signal
25 stift	9 stift		9 stift	
3	2	←	3	Tx
2	3	→	2	Rx
7	5	↔	5	GND
4	7	→	8	CTS (PC ej redo)
5	8	←	4	RTS (Maskin ej redo)

När maskinen sätts på startar den med grundinställningar enligt följande: 9600 Baud, ingen paritet, 8 databitar och en stoppbit. Överföringshastigheten ställs sedan normalt om till 115200 Baud.

Handskakningssignalerna CTS (Clear To Send) och RTS (Request To Send) används för att få kontroll över flödet. RTS används av maskinen för att indikera när den inte är redo att ta emot data från RX. Skickas något när RTS är aktiv, kan detta resultera i kommunikationsfel. För att undvika detta kan datorn tala om för maskinen att den inte får skicka något. Detta utförs genom att sätta CTS aktiv.

Ett mjukvaruprotokoll för konfigurering eller "setup" används för att kommunicera mellan dator och maskin. En allmän benämning på detta protokoll är PCcom. Alla meddelanden som skickas mellan dator och maskin har samma uppbyggnad:

25	Riktning	Data
	PC - Maskin	
	→	Escape-tecken (ESC 27)
	→	Kommando-tecken ('A' - 'Z')
	→	Underkommando-tecken ('a' - 'z' eller 'A' - 'Z')
30	↔	Data-bytes
	↔	CSUM, enkel byte-checksumma
	←	ACK (6) om CSUM är ok
		NAK (21) om CSUM inte är ok

EM (25) om CSUM är OK och det inte finns  
tillräckligt med minne för data.

\* \* \*

5 Med hänvisning åter till FIG 7 samt till FIG 8  
kommer nu den portabla anordningen 200 att beskrivas  
närmare. Därefter beskrivs dess arbetssätt med hänvisning  
till FIG 9-11.

Den portabla anordningen 200 strömförsörjs från  
10 maskinen 100 respektive datorn 120 via utgångarna på  
serieportarna 210, 220, enligt vad som visas i FIG 8.  
Serieportar används normalt enbart till kommunikation,  
varför det är viktigt att inte förstöra kommunikations-  
signalerna. För att skilja kommunikationssignalen 802  
15 från strömförsörjning har dioder 804 arrangerats som  
karaktäristiskt endast släpper igenom spänning i en  
riktning. Genom en spänningsdubblare 806 (t ex National  
Semiconductor LMC7660) kan en positiv spänning erhållas  
ur en negativ spänning för strömförsörjning 808 av anord-  
20 ningens komponenter. Strömförsörjningen 808 kan bl a  
innefatta en spänningsregulator såsom ST Microelectronics  
LE33ABZ för omvandling till lämplig spänningsnivå för  
komponenterna, t ex 3,3 V.

COM-portens spänningsområde ligger enligt RS232-  
25 specifikationen på  $\pm 25V$ . Spänningsområdet är emellertid  
inte lika i alla maskiner. Därför kan det finnas behov av  
att använda skyddsmotstånd på ingångssidan.

Styrenheten 700 kan realiseras av en mikrostyrkrets  
MSP430F149 från Texas Instruments. Denna mikrostyrkrets  
30 har en mycket låg strömförbrukning. Den drar endast  
250  $\mu A$  per MIPS vid 2,2 V matning och är i första hand  
framtagen för tillämpningar med hårda krav på låg energi-  
förbrukning. Matningsområdet ligger mellan 1,8 och 3,6 V.  
Vid en arbetsfrekvens på 5 MHz fås en total strömförbruk-  
35 ning för mikrostyrkretsen på ca 1300  $\mu A$ . Maximal arbets-  
frekvens är 8 MHz vid 3,6 V. Arkitekturen är uppbyggd  
på en 16-bitars RISC-kärna med 27 instruktioner. Varje

instruktion tar endast en klockcykel. Instruktions tiden vid 5 MHz blir då 0,2  $\mu$ s.

Mikrostyret har ett programminne av flashtyp på 32-60kB och ett arbetsminne av RAM-typ på 1-2 KB.

- 5 Dessutom har den en AD-omvandlare som med fördel används till spänningsövervakning och temperaturmätning.

En fördel med mikrostyret MSP430F149 och dess syskon i 14x-serien är den har två USART-kretsar ("Universal Synchronous Asynchronous Receive Transmit").  
 10 I den föredragna utföringsformen realiseras således kommunikationsenheterna 740 och 750 av dessa USART-kretsar och ingår med andra ord fysiskt i styrenheten 700.

Givetvis kan styrenheten 700 realiseras av någon annan komponent än den ovan föreslagna, exempelvis Atmel  
 15 ATMEGA161L 8PI 0143, Atmel T89C51RD2 eller Microchip PIC18LF452.

Vid realisering av minnet 710 kan man lämpligen beakta parallellprogrammerade flashminnen eller seriellt programmerade EEPROM. Fördelen med EEPROM är att de drar  
 20 lite ström vid skrivning jämfört med flashminnen. Flashminnen använder sig av "sektorer". Eftersom sektorn i flashminnet måste raderas innan skrivning kan ske, krävs att sektorn hämtas till styrenhetens RAM. RAM-minnet måste därför vara minst lika stort som flashminnets  
 25 sektorstorlek. I den föredragna utföringsformen valdes ett flashminne av typen SST SST28VF040A. Kapaciteten är 512 KB och matningsspänningsområdet 2,7 - 3,6 V. I den föredragna utföringsformen utnyttjas två fysiska minneskapslar, varvid kapaciteten totalt blir 1024 KB. Åtkomst-  
 30 tiden för minnet är 150-200 ns, vilket är kortare än åtkomsttiden hos den valda styrenheten. Minnet går att skriva över 100 000 ggr, och informationen ska hålla i 100 år. Alternativa val av minne 710 är exempelvis Microchip 24LC515, Microchip 25LC640, ST Microelectronics  
 35 M95256-V eller Atmel AT29BV040A.

Åtminstone delar av minnet 710 kan realiseras av internt minne i styrenheten, exempelvis programminnet



(flash) och arbetsminnet (RAM) i mikrostyret MSP430F149.

Den portabla anordningen drivs av programvara enligt följande. Programvarans arbetssätt framgår av FIG 9 och är uppbyggd enligt genomströmningsprincipen. Den startar med en initiering för att sedan gå in i en superloop. Här "snurrar" programmet flera tusentals varv per sekund. Omloppshastigheten beror på vad som händer i det aktuella skedet. Superloopen måste vara snabbare än den begränsande faktorn, vilken i detta fall är kommunikationshastigheten. Programmet skulle kunna betraktas som en form av tillståndsmaskin.

Varje "block" låser aldrig upp exekveringen i form av oändlighetsloopar utan återgår hela tiden till superloopen. Ett undantag är om något fel på det externa minnet skulle uppenbaras under initieringen. Det är olämpligt att initieringen i sådana fall fortsätter, eftersom resterande kod är totalt beroende av att det externa minnet fungerar.

Superloopens uppgift är att varje varv övervaka de olika mjukvarublockens tillstånd. Superloopen är hjärnan i mjukvaran och står högst i hierarkin. Den bestämmer bland annat när det är dags att skriva till externminnet eller när det är dags att kommunicera. Till sin hjälp har den mjukvarubuffertar, tidmättningsfunktioner, avbrottsfunktioner, funktionspekare och kommunikationsenheter. Det finns även övervakningsklockor som hela tiden kontrollerar att allt står rätt till.

Mottagningen av information från datorn sker med hjälp av ett avbrott. Avbrottet fyller en mellanlagringsbuffert som har plats för 20 byte. En cirkulär skrivpekare som ägs av avbrottet används för att placera data i mellanlagringsbufferten. Cirkulär innebär att den inte alltid startar lagringen från adress noll utan fortsätter lagra från dess nuvarande position. När den når slutet förflyttar den sig till position noll igen. Vid varje nytt datapaket räknas en räknare upp med ett. Denna

räknare håller reda på hur mycket som är lagrat i mellanlagringsbufferten och ägs av både avbrottet och superloopen. Räknaren kontrolleras av superloopen varje varv. Då räknaren skiljer sig från noll aktiveras en PCcom-funktion (jfr vad som beskrivits tidigare i detta dokument). PCcom tolkar informationen med hjälp av en läspekare och avgör om den är en del av ett ESC-kommando eller data. Data placeras med hjälp av en annan skrivpekare i en databuffert på 129 byte. Läspekaren är cyklisk och ägs av superloopen. Skrivpekaren till databufferten är däremot inte cyklisk utan startar från position noll vid varje ny lagringssekvens. Denna pekare ägs av superloopen. Data finns nu upplagda i databufferten och är redo att placeras vidare av PCcom. PCcom kan peka direkt på önskad databyte.

För att förenkla programmeringen används två separata mjukvarubuffertar, en för varje avbrott. Mjukvarubuffertarna är tilldelade minsta möjliga storlek utan att kommunikationen försämras. Anledningen är att de tar upp plats i RAM-minnet som endast är exempelvis 2 KB. Eftersom datorn skickar max 129 byte data per sekvens, är detta en lämplig storlek på databufferten till PCcom.

Mottagning från maskinen 100 sker enligt samma princip som mottagning från datorn 120. Skillnaden är att den portabla anordningen 200 använder en annan mellanlagringsbuffert och annan databuffert. Dessa buffertar har en annan storlek. Mellanlagringsbufferten är på 15 byte och databufferten är på 10 byte.

Sändning till datorn 120 och maskinen 100 sker via en gemensam mjukvarubuffert på 920 byte. Sändningen är inte avbrottsstyrd utan sköts av superloopen. Superloopen håller reda på var data ska skickas med hjälp av flaggor.

Extern minneshantering, dvs hantering av data till och från det externa minnet, sker via funktioner. För att skriva till minnet anropas en skivfunktion med inparametrarna data och adress. Vid läsning anropas en läsfunktion med inparametern adress, varvid data returneras.

De två minneskapslarna är sammanslagna till ett block med hjälp av mjukvara på låg nivå. Utåt presenteras detta som ett stort sammanhängande minne med linjär adressering från 0x00000-0xFFFFF, dvs 1 MB.

5           Exempel:

Byte a och b programmeras på adresserna x och y (a och b placeras i olika sektorer). Följande sker:

1) Den sektor som innehåller adress x kopieras från externminnet till minneshanteringsbufferten i mikro-  
10 styrkretsens RAM. Detta sker inte om aktuell sektor redan finns i RAM.

2) Byte a placeras på adress x i minneshanteringsbufferten.

3) Sektor innehållande adress x i externminnet  
15 raderas.

4) Minneshanteringsbufferten programmeras till aktuell sektor i externminnet.

5) Den sektor som innehåller adress y kopieras från externminnet till minneshanteringsbufferten.

20 6) Byte b placeras på adress y i minneshanteringsbufferten.

Byte b programmeras inte till externminnet förrän sektorbyte eller "timeout" sker.

\* \* \*

25 I programflödet enligt FIG 9 sker först initiering (steg 900-920), varefter superloopen inträds (steg 922-938). Under initieringen sker följande:

Programmet startar med att initiera alla globala  
variabler i steg 902. CPU:ns yttre klockfrekvens (X1)  
30 ställs in tills oscillatorn stabiliserats. Den inre klockan (DCO) ställs in till 500 kHz. Portarnas riktning ställs in.

Därefter kontrollerar programmet om det externa flashminnet är OK i steg 904. Om minnet inte är OK, indikeras ett felmeddelande i steg 906 med hjälp av lysdiod-  
35 erna 720/230-232. Om minnet däremot är OK, fortsätter programmet med de sista initieringarna i steg 908:

AD-omvandlaren ställs in att omvandla 15 ggr per sekund. En timer initieras till att ge avbrott var 100:ems. Överföringshastigheten ställs in till 9600 Baud. Realtidsklockan (RTC) tilldelas en godtycklig tid och datum.

Därefter sätts det globala avbrottet på i steg 910. I steg 912 kontrollerar programmet om det finns några konfigurationsdata lagrade i det externa minnet. Om så inte är fallet programmeras minnet med en grundkonfiguration i steg 914.

I stegen 916 och 918 ges maskin och dator tillstånd att sända. Slutligen startas timern till RTC i steg 920. Därmed är initieringsdelen avslutad.

I superloopen uppdateras först RTC i steg 922. Det kontrolleras i stegen 924 och 926 huruvida eventuell timeout har inträffat respektive vilket meddelande som ska visas. Därefter ser programmet i steg 928 efter vad som är finns anslutet till kommunikationsenheterna. Följande tre fall kan inträffa:

- Endast datorn är ansluten (steg 932, FIG 11).
- Endast maskinen är ansluten (steg 930, FIG 10).
- Både datorn och maskinen är anslutna (steg 934-936).

Då endast datorn är ansluten sker följande enligt FIG 11. Maskinens UART-drivkrets stängs av i steg 1100, och datorns UART-drivkrets sätts på i steg 1102. Det kontrolleras i steg 1104 om något nytt tecken har kommit från datorn. Om så är fallet körs PCcom i steg 1106. Detta steg har till uppgift att tolka inkommande data från datorns konfigureringsprogram på samma sätt som maskinen skulle gjort ifall maskinen vore ansluten direkt till datorn. Efter detta kontrolleras i steg 1108 om något tecken behöver skickas till datorn. Till slut kontrolleras i steg 1110 om datorn begärt ändrad överföringshastighet.

Då endast maskinen är ansluten sker följande enligt FIG 10. För att förenkla användandet av den portabla

anordningen tilldelas olika uppgifter som lagras i "status\_mode". Programmet kontrollerar "status\_mode" och utför given uppgift som beror på aktuell "status\_mode". Det finns totalt åtta olika sätt att kombinera

5 "status\_mode":

· Ingen "status\_mode" är vald (steg 1002, 1004): Exekveringen avslutas. I annat fall stängs kommunikationsenheten 740 mot datorn av i steg 1006, medan kommunikationsenheten 750 mot maskinen sätts på i steg 1008.

10 · Uppdatera maskinens programvara (steg 1010, 1012): Inloggning på maskinen enligt ett fördefinierat programuppdateringsprotokoll. Efter uppdatering sker mjukvaruomstart av maskinen i steg 1014. Om "status\_mode" indikerar ytterligare något utöver programuppdatering  
15 (kontrolleras i steg 1015), fortsätter exekveringen med inloggning på maskinen enligt ett fördefinierat konfigureringsprotokoll (steg 1016).

· Ladda upp konfigureringsdata: Uppdatera konfigureringsdata i maskinen (steg 1018, 1020 resp 1024).

20 · Databandspelare: Spela in driftdata från maskinen och lagra i det externa flashminnet (steg 1018, 1022 resp 1026). Inspelade driftdata kan sedan överföras till datorn för analys eller liknande.

Då både datorn och maskinen är anslutna sätts bägge  
25 kommunikationsenheter 740, 750 på i steg 934 och 936. I detta läge är den portabla anordningen 200 genomskinlig för datorn och maskinen. Dess uppgift är endast att lagra viktig information samt skicka vidare all inkommande information.

30 Uppfinningen har beskrivits ovan i form av ett par exemplifierande utföringsformer. Dock är uppfinningen inte på något vis begränsad till dessa utan inrymmer många andra varianter, enligt vad som definieras av skyddsomfånget hos de bifogade patentkraven och vidare  
35 enkelt inses av fackmannen. En möjlighet i sammanhanget är att utnyttja en och samma fysiska kommunikationsport och/eller kommunikationsenhet hos den portabla anordning-

en för såväl anslutningen mot datorn som för anslutningen  
mot maskinen. I en sådan utföringsform får styrenheten i  
den portabla anordningen arrangeras att känna igen vilken  
av datorn eller maskinen som för närvarande är ansluten  
5 och agera i beroende därav.

PATENTKRAV

1. Portabel anordning (200) för konfigurering av  
 5 en maskin (100) av sådan typ som hanterar mynt eller  
 värdepapper och som har ett gränssnitt (402) för an-  
 slutning av en extern enhet (120), varigenom maskinen  
 kan konfigureras från den externa enheten, varvid den  
 portabla anordningen **kännetecknas av**  
 10 en styrenhet (700);  
 ett minne (710);  
 ett första gränssnitt (210, 740) för anslutning  
 till nämnda externa enhet (120); samt  
 ett andra gränssnitt (220, 750) för anslutning  
 15 till gränssnittet (402) på nämnda maskin (100);  
 varvid styrenheten är anordnad, då den portabla  
 anordningen är ansluten till den externa enheten via  
 det första gränssnittet, att emulera nämnda maskin  
 samt att från den externa enheten ta emot och i minnet  
 20 lagra konfigureringsdata avsedda för nämnda maskin,  
 och  
 varvid styrenheten är anordnad, då den portabla  
 anordningen är ansluten till nämnda maskin via det  
 andra gränssnittet, att emulera den externa enheten  
 25 samt att till nämnda maskin avge nämnda i minnet  
 lagrade konfigureringsdata.

2. Portabel anordning enligt krav 1, varvid den  
 externa enheten (120) är en dator av typen bärbar dator,  
 30 persondator, handdator, arbetsstation eller liknande.

3. Portabel anordning enligt krav 1 eller 2, varvid  
 de första och andra gränssnitten (210, 740, 220, 750) är  
 seriella.  
 35

4. Portabel anordning enligt något av ovanstående  
 krav, varvid nämnda konfigureringsdata innefattar åtmin-

stone något av följande: inställningar och/eller parametrar för giltighet/äkthet/valör/typ för mynt respektive värdepapper; referensdata; eller data avseende språkstöd.

5           5. Portabel anordning enligt något av ovanstående krav, varvid nämnda konfigureringsdata innefattar programvara för exekvering i en i maskinen (100) ingående mikroprocessor (632).

10           6. Portabel anordning enligt något av ovanstående krav,

          varvid styrenheten (700) är anordnad, då den portabla anordningen (200) är ansluten till nämnda maskin (100) via det andra gränssnittet (210, 750),  
15       att ta emot driftdata från nämnda maskin och lagra dessa i minnet (710); samt

          varvid styrenheten (700) är anordnad, då den portabla anordningen (200) är ansluten till den externa enheten (120) via det första gränssnittet  
20       (210, 740), att till den externa enheten överföra nämnda i minnet lagrade driftdata.

          7. Portabel anordning enligt krav 6, varvid nämnda driftdata innefattar antal/giltighet/äkthet/  
25       valör/typ för mynt respektive värdepapper som hanterats av maskinen (100), och/eller av maskinen detekterade fysikaliska parametrar för sådana mynt respektive värdepapper.

30           8. Portabel anordning enligt krav 3, varvid anordningen erhåller strömförsörjning från de första och andra seriella gränssnitten (210, 740, 220, 750).

35           9. Portabel anordning enligt något av ovanstående krav, varvid det första och andra gränssnittet utgörs av en och samma fysiska enhet.



10. Portabel anordning enligt något av ovanstående krav, varvid nämnda maskin (100) är en myntsorterare eller mynträknare.

- 5           11. Metod för konfigurering av en maskin (100) av sådan typ som hanterar mynt eller värdepapper och som kan konfigureras från en extern enhet (120), varvid metoden **kännetecknas av** stegen att
- 10           ansluta en portabel anordning (200), annan än nämnda maskin (100) och nämnda externa enhet (120), till nämnda externa enhet (120);
- från den externa enheten ta emot och i den portabla anordningen lagra konfigureringsdata avsedda för nämnda maskin;
- 15           ansluta den portabla anordningen till nämnda maskin; samt
- till nämnda maskin avge nämnda lagrade konfigureringsdata.
- 20           12. Metod enligt krav 11, varvid den externa enheten (120) är en dator av typen bärbar dator, persondator, handdator, arbetsstation eller liknande.

- 25           13. Metod enligt krav 11 eller 12, varvid nämnda konfigureringsdata innefattar åtminstone något av följande: inställningar och/eller parametrar för giltighet/äkthet/valör/typ för mynt respektive värdepapper; referensdata; eller data avseende språkstöd.

- 30           14. Metod enligt något av kraven 11-13, varvid nämnda konfigureringsdata innefattar programvara för exekvering i nämnda maskin (100).

- 35           15. Metod enligt något av kraven 11-14, vidare innefattande stegen att
- ansluta den portabla anordningen (200) till nämnda maskin (100);

ta emot driftdata från nämnda maskin och lagra dessa i den portabla anordningen;

ansluta den portabla anordningen till den externa enheten (120); samt

5 till den externa enheten överföra nämnda lagrade driftdata.

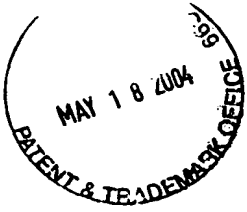
16. Metod enligt krav 15, varvid nämnda drift-  
data innefattar antal/giltighet/äkthet/valör/typ för  
10 mynt respektive värdepapper som hanterats av maskinen  
(100), och/eller av maskinen detekterade fysikaliska  
parametrar för sådana mynt respektive värdepapper.

17. Metod enligt något av kraven 11-16, varvid  
15 nämnda maskin (100) är en myntsorterare eller mynt-  
räknare.

## SAMMANDRAG

En portabel anordning (200) presenteras för  
5 konfigurerings av en maskin (100) av sådan typ som  
hanterar mynt eller värdepapper och som har ett gränssnitt (402) för anslutning av en extern enhet (120),  
varigenom maskinen kan konfigureras från den externa  
enheten. Den portabla anordningen har en styrenhet  
10 (700); ett minne (710); ett första gränssnitt (210,  
740) för anslutning till nämnda externa enhet (120);  
samt ett andra gränssnitt (220, 750) för anslutning  
till gränssnittet (402) på nämnda maskin (100).  
Styrenheten är anordnad, då den portabla anordningen  
15 är ansluten till den externa enheten via det första  
gränssnittet, att emulera nämnda maskin samt att från  
den externa enheten ta emot och i minnet lagra kon-  
figureringsdata avsedda för nämnda maskin. Styrenheten  
är vidare anordnad, då den portabla anordningen är an-  
20 sluten till nämnda maskin via det andra gränssnittet,  
att emulera den externa enheten samt att till nämnda  
maskin avge nämnda i minnet lagrade konfigurerings-  
data.

25 Att publiceras tillsammans med FIG 7.



1/8

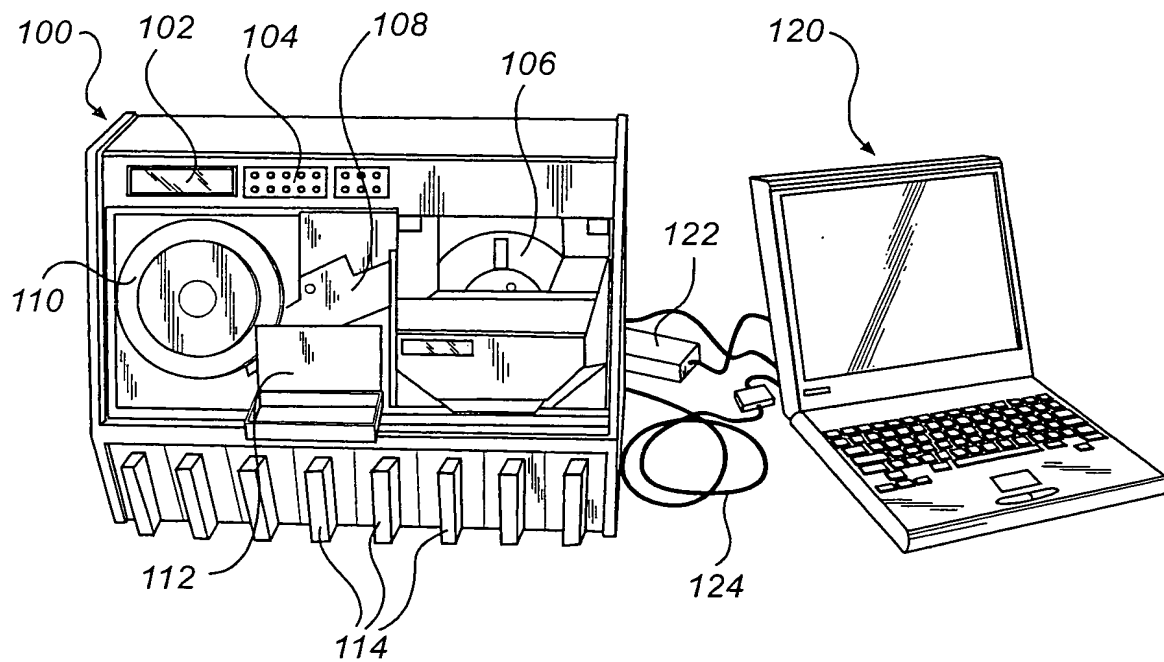


Fig. 1

FIG. 1

2/8

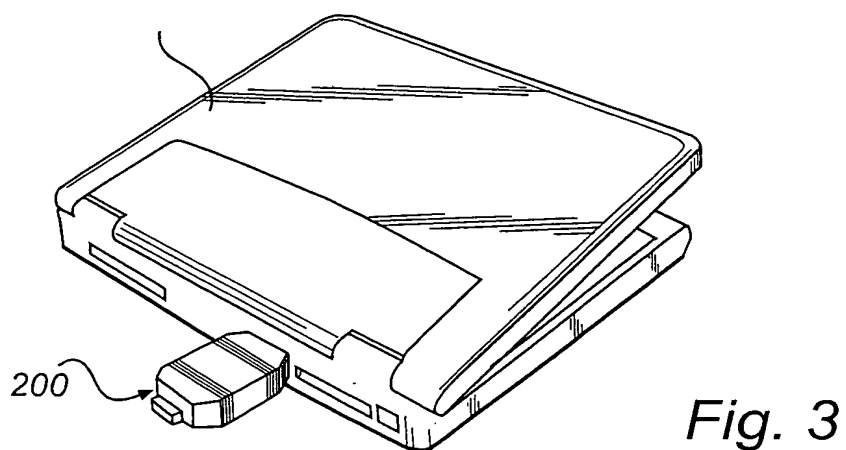
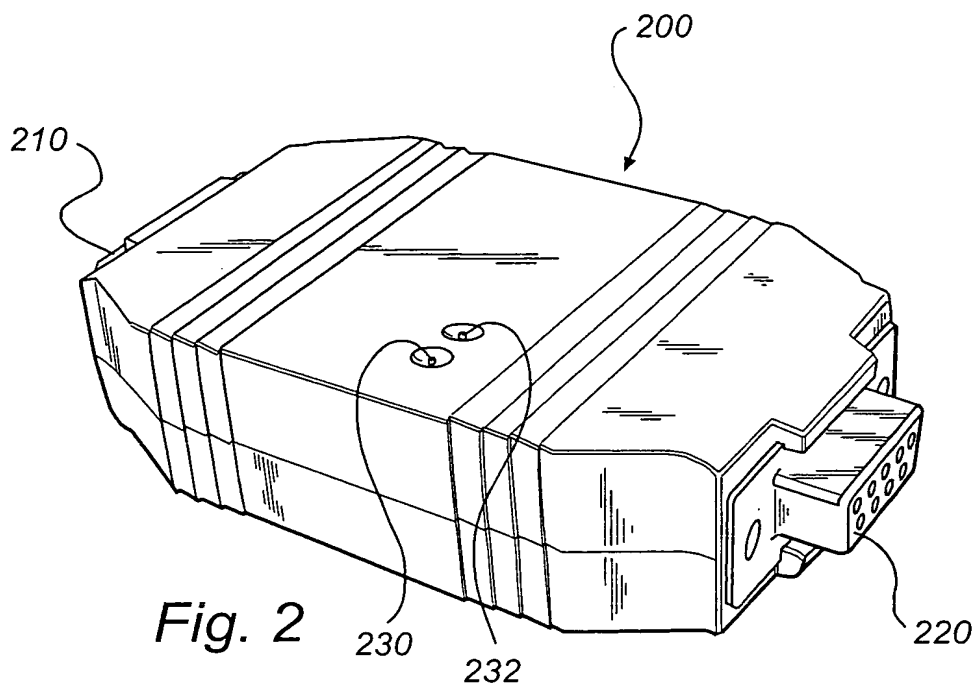
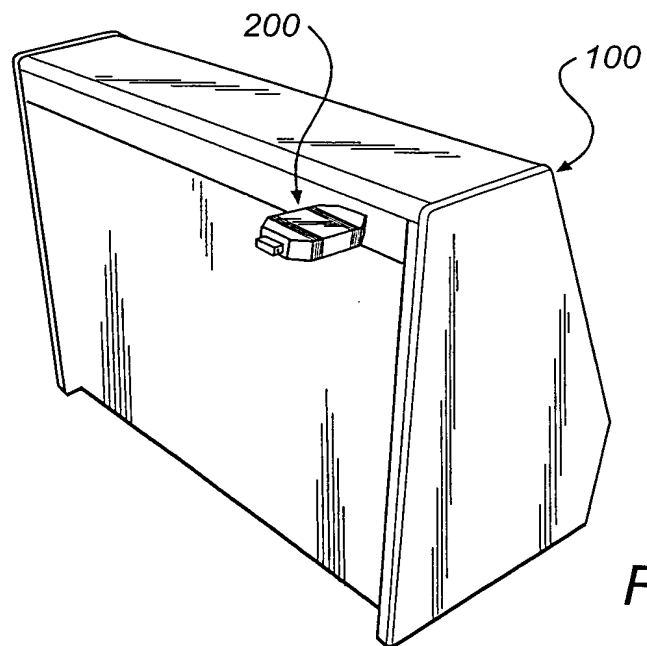
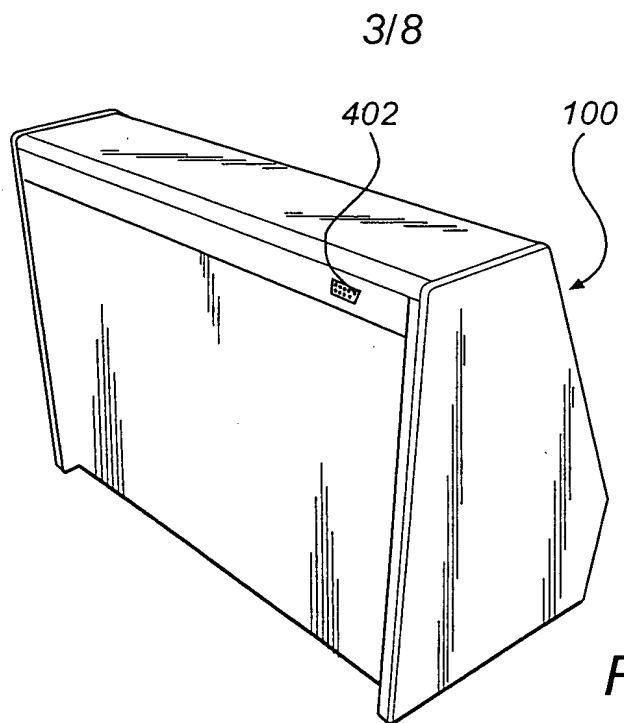


FIG. 3



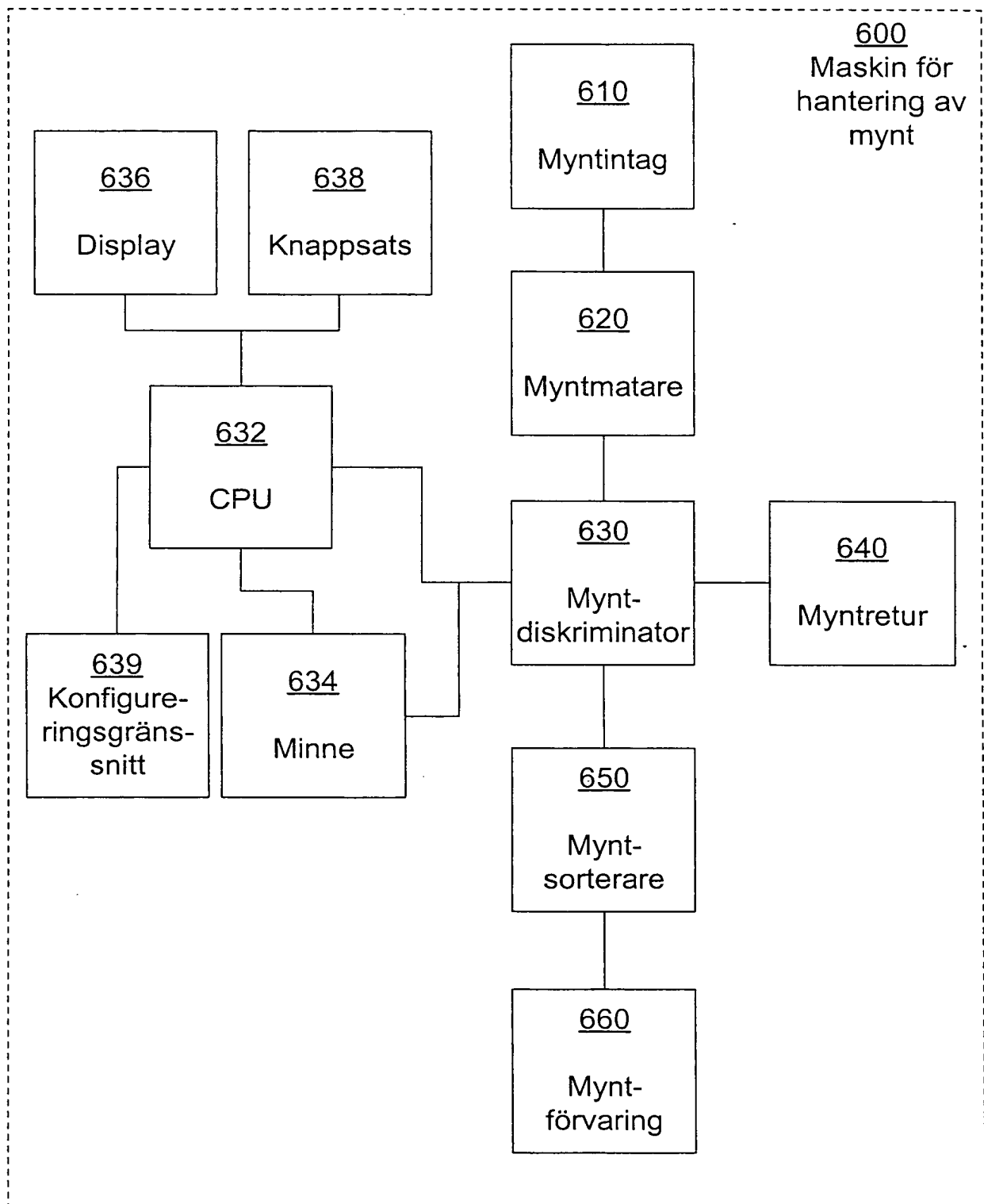


Fig 6

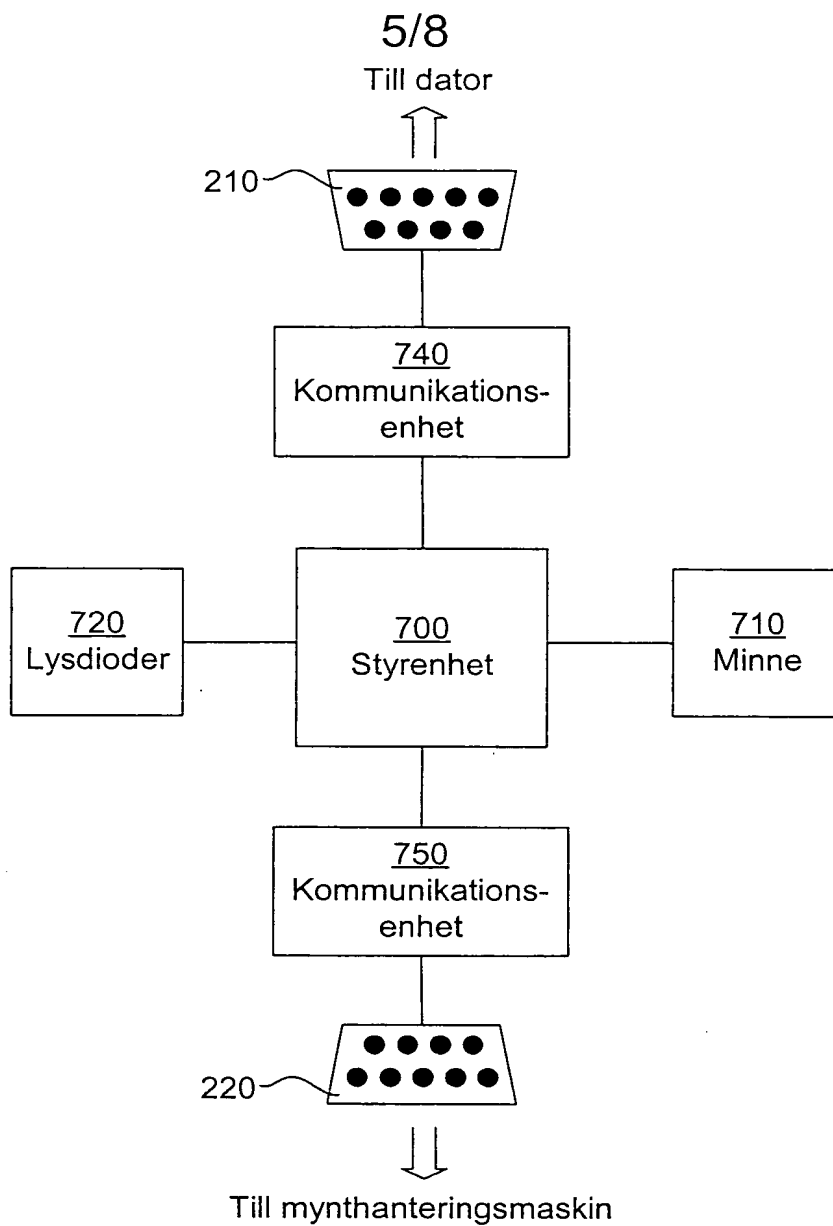


Fig 7

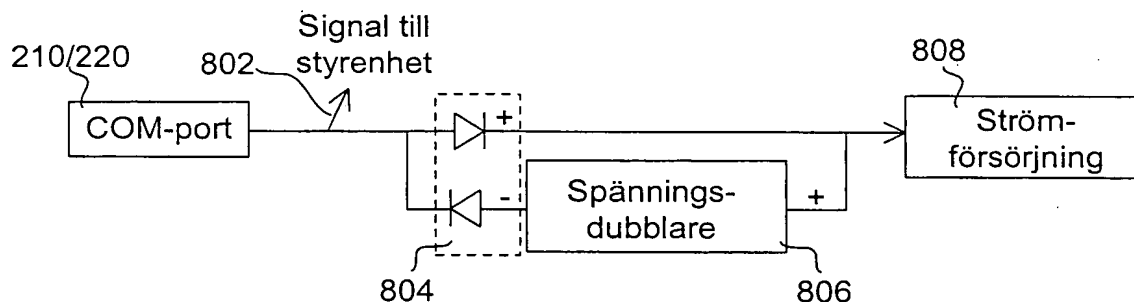


Fig 8



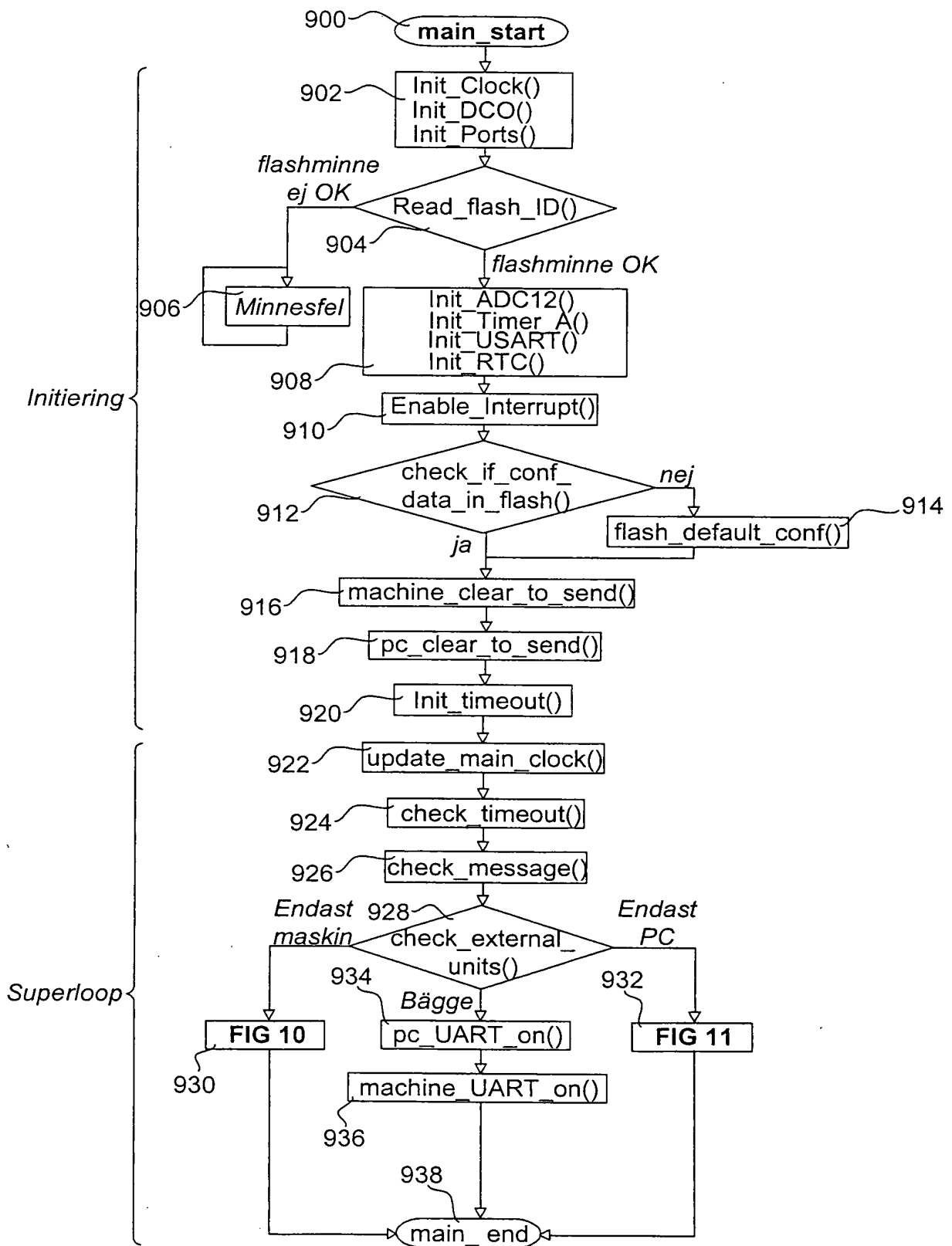


Fig 9

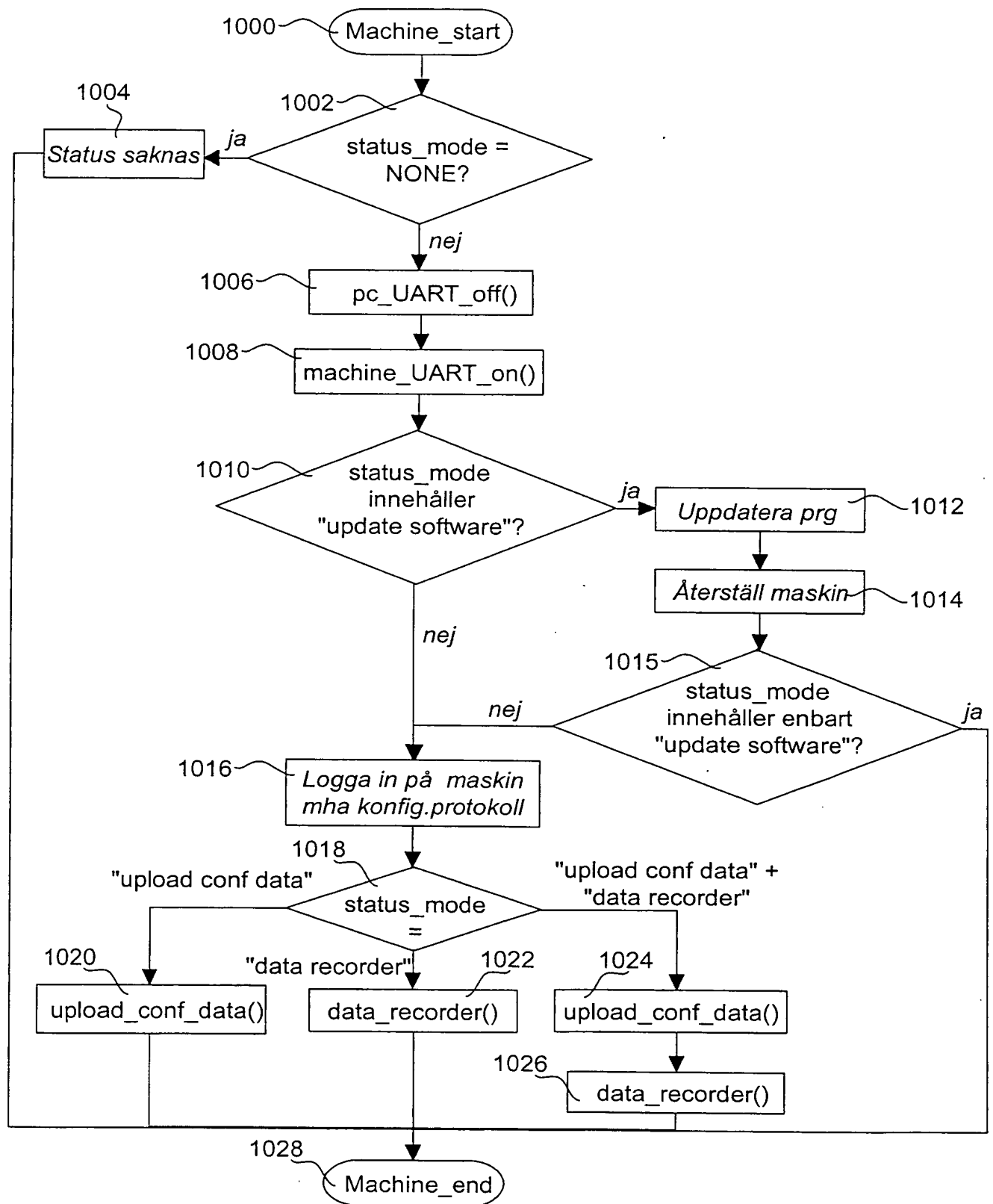
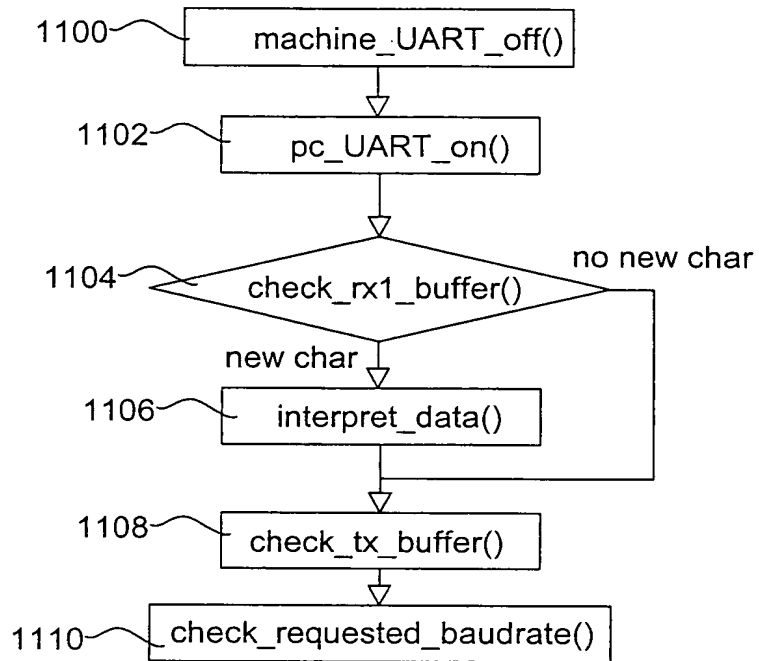


Fig 10

*Fig 11*